



Fachhochschule Köln
Cologne University of Applied Sciences

Physikalisch plausibles Rendering für Echtzeitanwendungen

Bachelorarbeit Medientechnik
Fachhochschule Köln
Institut für Medien- und Phototechnik

Tobias Bayer
Matrikelnummer: 11066471

Erstgutachter: Prof. Dr.-Ing. Arnulph Fuhrmann
Zweitgutachter: Prof. Dr. Stefan Grünvogel

06.08.2012



Fachhochschule Köln
Cologne University of Applied Sciences

Physically Plausible Rendering for Realtime Applications

Bachelor-Thesis in Media Technology
Cologne University of Applied Sciences
Institute for Media and Imaging Technology

Tobias Bayer

Matriculation-Number: 11066471

First Reviewer: Prof. Dr.-Ing. Arnulph Fuhrmann

Second Reviewer: Prof. Dr. Stefan Grünvogel

08/06/2012

Abstract

Titel: Physikalisch plausibles Rendering für
Echtzeitanwendungen

Autor: Tobias Bayer

Referent: Prof. Dr.-Ing. Arnulph Fuhrmann

Korreferent: Prof. Dr. Stefan Grünvogel

Zusammenfassung: Diese Arbeit behandelt die physikalisch
plausible Bildsynthese virtueller Szenen an-
hand strahlungsphysikalischer Größen sowie
physikalisch plausibler Beleuchtungsmodelle
in Kombination mit qualitativ hochwer-
tigem Schattenwurf. Verschiedene Modelle
und Verfahren werden untersucht und in
einer Echtzeitumgebung implementiert.

Schlüsselwörter: Rendering Beleuchtungsmodelle BRDF
Schatten Echtzeit

Datum: 06.08.2012

Abstract

Titel: Physically Plausible Rendering for Realtime Applications

Author: Tobias Bayer

First Reviewer: Prof. Dr.-Ing. Arnulph Fuhrmann

Second Reviewer: Prof. Dr. Stefan Grünvogel

Summary: This thesis covers the physically plausible rendering of virtual scenes using radiometric quantities and physically plausible illumination models in combination with high quality shadows. Multiple models and techniques are being evaluated and implemented into a realtime environment.

Keywords: Rendering Illumination Models BRDF Shadows Realtime

Date: 08/06/2012

Inhaltsverzeichnis

1	Einleitung	4
2	Grundlagen	5
2.1	Radiometrische Größen	6
2.1.1	Strahldichte	6
2.1.2	Bestrahlungsstärke	8
2.2	Die Rendergleichung	9
2.3	Dynamikkompression	11
2.4	Lineare Wiedergabe radiometrischer Größen	13
3	Lokale Beleuchtungsmodelle	16
3.1	Die bidirektionale Reflektanzverteilungsfunktion	16
3.2	Der Brechungsindex	20
3.3	Gerichtete Reflexion	23
3.3.1	Das Phong-Modell	23
3.3.2	Das Torrance-Sparrow-Modell	28
3.4	Diffuse Reflexion	32
3.4.1	Der Lambertsche Reflektor	32
3.4.2	Das Oren-Nayar-Modell	34
3.5	Mehrschichtige Oberflächen	36
4	Schatten	39
4.1	Shadow Mapping	40
4.2	Percentage Closer Filtering	44
4.3	Variance Shadow Maps	47
4.4	Exponential Variance Shadow Maps	50
4.5	Umgebungsverdeckung	53
5	Implementierung	55
6	Zusammenfassung und Ausblick	57
	Literatur	60

1 Einleitung

Das Gebiet der Computergrafik umfasst unter anderem die Bildsynthese einer virtuellen Szene. Diese Bildsynthese erfolgt durch einen mehr oder weniger komplexen Rendervorgang. Mit steigender Rechenleistung aktueller Grafikhardware werden immer anspruchsvollere Renderingtechniken echtzeitfähig.

Die plausible Darstellung realitätsnaher Materialien wird in Echtzeitanwendungen oftmals vernachlässigt. Um den Eindruck einer realen Szene durch die Visualisierung einer virtuellen Szene zu erwecken, müssen reale Parameter und Verhältnisse strahlungsphysikalischer Größen zur Beleuchtung der Szene verwendet werden.

Dabei ist neben der physikalisch plausiblen Interaktion des Lichtes auf den Oberflächen sowie des glaubhaften Schattenwurfs der Geometrie, auch die korrekte Reproduktion des Ergebnisses durch das Anzeigegerät von großer Bedeutung.

Das Ziel dieser Arbeit ist eine echtzeitfähige Implementierung physikalisch plausibler Beleuchtungsmodelle in Verbindung mit qualitativ hochwertiger Schattendarstellung.

Kapitel 2 erläutert die Grundlagen der Radiometrie, die realistische Bildsynthese anhand der Rendergleichung inklusive einer Vereinfachung für Echtzeitanwendungen sowie die lineare Wiedergabe eines gerenderten Bildes.

In Kapitel 3 werden verschiedene lokale Beleuchtungsmodelle sowohl zur Beschreibung diffuser als auch gerichteter Lichtreflexionen sowie die Kombination der verschiedenen Anteile betrachtet und implementiert.

Kapitel 4 befasst sich mit der Erzeugung von Verdeckungsinformationen zur Abschattung der Lichtquellen einer virtuellen Szene. Es werden mehrere Verfahren zur Verringerung von Aliasing-Effekten des Schattenwurfs verglichen.

Zuletzt wird in Kapitel 5 der Ablauf des implementierten Rendervorganges der im Rahmen dieser Arbeit entwickelten Software näher erläutert.

2 Grundlagen

Bei der Bildsynthese unterscheidet man generell zwischen Raytracing und Rasterisierung. Raytracing modelliert den Lichttransport im Raum und kann eine Reihe von Problemen in der Computergrafik lösen, unter anderem Lichtbrechung an Grenzflächen, Spiegelung, Schattierung sowie auch indirekte Beleuchtung. Diese Eigenschaften erfordern jedoch ein hohes Maß an Rechenleistung. Mit aktueller Hardware sind, je nach Komplexität der Szene, interaktive bis echtzeitfähige Bildwiederholraten bei lediglich moderaten Auflösungen möglich. Aufgrund der intensiven Berechnungen wird Raytracing selten im Echtzeit-Bereich verwendet, ist dennoch aber der Standard für hochqualitatives Offline-Rendering wie es unter anderem in der Film- und Werbeindustrie benötigt wird.

Rasterisierung ermöglicht im Gegenzug eine hocheffiziente Darstellung von geometrischen Primitiven wie Punkten, Linien oder Dreiecken mithilfe von Hardwarebeschleunigung. Häufig vorkommende Berechnungen wie die Transformation der Geometrie, Interpolation von Geometrieattributen sowie letztendlich auch die Rasterisierung werden direkt in Hardware durch die Grafikkarte ausgeführt. Um schließlich realitätsnahe Bilder zu generieren, müssen allerdings eine Reihe verschiedener Techniken zur Beleuchtung und Schattierung eingesetzt werden. Diese Techniken werden in den weiteren Kapiteln näher veranschaulicht.

Alle gerenderten Bilder dieser Arbeit wurden, soweit nicht anders vermerkt, mit der im Rahmen dieser Arbeit entwickelten Software erstellt.

2.1 Radiometrische Größen

Die visuelle Wahrnehmung beschreibt die Reaktion auf visuelle Reize. Diese visuellen Reize werden bei Menschen durch den Photonentransport in Form von elektromagnetischer Strahlung mit Wellenlängen von etwa 380 nm bis 780 nm ausgelöst [PPB05]. Die Energie elektromagnetischer Strahlung pro Zeiteinheit ist die Strahlungsleistung $\Phi[\text{W}]$.

Anstelle von spektralen Größen werden in der Computergrafik überwiegend RGB-Werte verwendet. Das Spektrum einer strahlungstechnischen Größe kann mithilfe der CIE-Spektralwertfunktionen in einen RGB-Farbvektor konvertiert werden [Ill04]. Durch das visuelle System des Menschen erzeugt ein solcher Farbvektor den Farbreiz der ursprünglichen spektralen Größe.

Die Radiometrie ist das wissenschaftliche Gebiet der Messung elektromagnetischer Strahlung. Für die Berechnung der Ursache der visuellen Wahrnehmung ist also ein Grundverständnis radiometrischer Größen unabdingbar. Dabei sind zwei Größen in der Computergrafik besonders maßgebend, die *Strahldichte* (engl. *Radiance*) und die *Bestrahlungsstärke* (engl. *Irradiance*).

2.1.1 Strahldichte

Die Strahldichte ist diejenige Größe, die durch den Sensor einer Kamera, oder auch das menschliche Auge, gemessen wird. Die realitätsnahe Bildsynthese ist daher die Visualisierung der Strahldichte von einem Punkt in einer Szene in Richtung des Betrachters.

Da sich elektromagnetische Strahlung im Raum ausbreitet, bezieht sich diese Richtungsabhängigkeit auf einen *Raumwinkel*. Ein Raumwinkel ω beschreibt einen Raumbereich und ist definiert als

$$\omega = \frac{A}{r^2}[\text{sr}]. \quad (2.1)$$

Die Einheit Steradian ist dimensionslos und entspricht dem Verhältnis einer Fläche A auf der Oberfläche einer Kugel zu dem Quadrat des Radius r dieser Kugel (Abbildung 2.1).

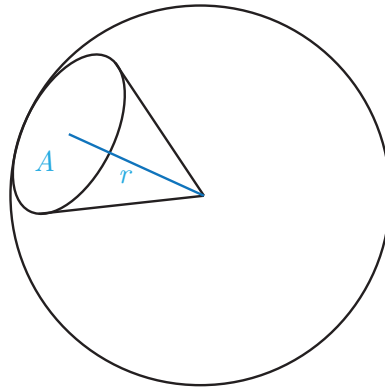


Abbildung 2.1 Veranschaulichung des Raumwinkels

Der Raumwinkel der Hemisphäre über einem Punkt, ausgerichtet nach der Oberflächennormalen des Punktes, wird im weiteren Verlauf Ω genannt.

Die Definition der Strahldichte L ist

$$L = \frac{d^2\Phi}{dA_{proj}d\omega} [\text{W m}^{-2} \text{sr}^{-1}]. \quad (2.2)$$

Die Strahldichte L ist die Strahlungsleistung $d^2\Phi$ [W] in Relation zu der in Abstrahlrichtung projizierten Fläche $dA_{proj} = dA \cos \theta$ [m²] mit $\cos \theta \geq 0$, sowie dem Raumwinkel $d\omega$ [sr] (Abbildung 2.2).

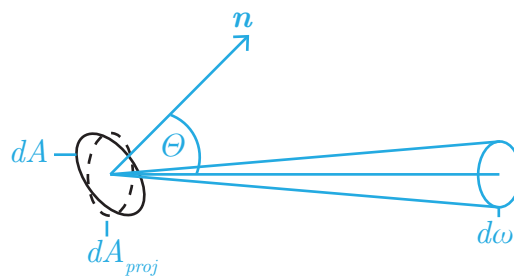


Abbildung 2.2 Die geometrischen Bestandteile der Strahldichte

2.1.2 Bestrahlungsstärke

Die Bestrahlungsstärke beschreibt die gesamte, auf der Oberfläche einer Geometrie eintreffende Strahlungsleistung pro Flächeneinheit. Sie ist definiert als

$$E = \frac{d\Phi}{dA} [\text{W m}^{-2}]. \quad (2.3)$$

Die Bestrahlungsstärke E kann aus der Strahldichte L in einem Punkt durch Multiplikation mit dem durch die abstrahlende Fläche aufgespannten Raumwinkelement ω_f , unter Berücksichtigung des Einfallswinkels θ aufgrund des Lambertschen Gesetzes (Abbildung 2.3) [LA92], ermittelt werden:

$$E = \int_{\Omega} L \cos \theta d\omega_f \quad (2.4)$$

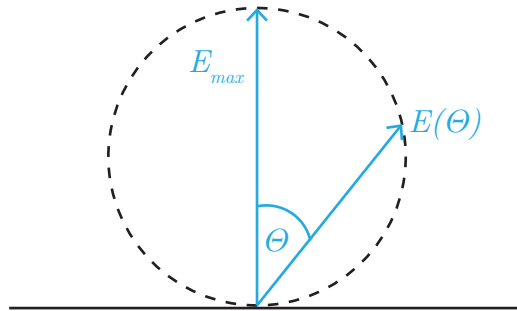


Abbildung 2.3 Die Bestrahlungsstärke ist abhängig von dem Einfallswinkel θ

Für ideale Punktlichtquellen lässt sich die Bestrahlungsstärke aufgrund der unendlich geringen Oberfläche jedoch nicht aus der Strahldichte ermitteln. Somit ergibt sich die Bestrahlungsstärke in einem Punkt durch eine Punktlichtquelle in Abstand d mit $A = 4\pi d^2$ unter Berücksichtigung des Einfallswinkels θ mit $\cos \theta \geq 0$ zu

$$E = \frac{\Phi}{4\pi d^2} \cos \theta. \quad (2.5)$$

Die Bestrahlungsstärke nimmt bei der Beleuchtung durch ideale Punktlichtquellen also proportional zum Quadrat der Entfernung ab [PPB05].

2.2 Die Rendergleichung

Erstmals 1986 von Kajiya vorgestellt [Kaj86], ist die Rendergleichung, basierend auf dem Energieerhaltungssatz, bis heute die mathematische Grundlage für alle realistischen Beleuchtungsmodelle in der Computergrafik. Gleichung 2.6 ist die äquivalente, moderne Formulierung der ursprünglichen Form [PH10].

Rendergleichung:

$$L_o(\mathbf{p}, \omega_o) = L_e(\mathbf{p}, \omega_o) + \int_{\Omega} f_r(\mathbf{p}, \omega_i, \omega_o) L_i(\mathbf{p}, \omega_i) \cos \theta_i d\omega_i \quad (2.6)$$

$L_o(\mathbf{p}, \omega_o)$:	gesamte Strahldichte von \mathbf{p} in Richtung ω_o
$L_e(\mathbf{p}, \omega_o)$:	emittierte Strahldichte von \mathbf{p} in Richtung ω_o
$f_r(\mathbf{p}, \omega_i, \omega_o)$:	bidirektionale Reflektanzverteilungsfunktion
$L_i(\mathbf{p}, \omega_i)$:	Strahldichte in \mathbf{p} aus Richtung ω_i
θ_i :	Winkel zwischen ω_i und der Oberflächennormalen in \mathbf{p}

Nach dieser Gleichung ist die Strahldichte L_o , von einem Oberflächenpunkt \mathbf{p} in Richtung ω_o , die Summe der emittierten und der reflektierten Strahldichte. Der reflektierte Anteil ist die aus allen Richtungen ω_i auf der Hemisphäre von \mathbf{p} in Richtung ω_o reflektierte Strahldichte, gewichtet durch die bidirektionale Reflektanzverteilungsfunktion (Kapitel 3.3) und dem Kosinus des Winkels zwischen ω_i und der Oberflächennormalen. Das Rendern einer Szene kann als das Lösen der Rendergleichung verstanden werden.

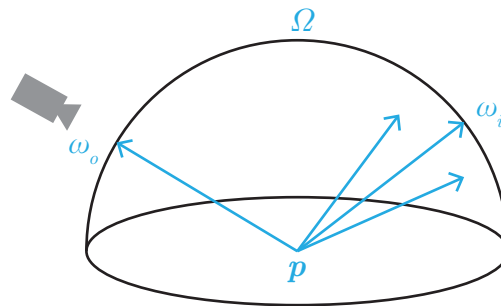


Abbildung 2.4 Veranschaulichung der Parameter der Rendergleichung

Obwohl die Gleichung sehr allgemein gehalten ist, werden nicht alle Aspekte des Lichttransports erfasst. Unter anderem werden Phosphoreszenz, Fluoreszenz, Interferenz sowie Lichtstreuung in einem Medium vernachlässigt.

In der Regel ist die Rendergleichung dennoch zu komplex um in praktikabler Zeit gelöst zu werden, daher müssen besonders für Echtzeit-Anwendungen einige Vereinfachungen durchgeführt werden. In der im Rahmen dieser Arbeit entwickelten Software wird lediglich die direkte Beleuchtung betrachtet, indirekte Beleuchtung wird vernachlässigt. Je nach Ausleuchtung einer Szene fällt dies mehr oder weniger stark auf. Die indirekte Beleuchtung liegt auf Grund ihres Ausmaßes außerhalb der Reichweite dieser Arbeit.

Weiterhin wird angenommen, dass sämtliche Lichtquellen einer Szene ideale Punktlichtquellen darstellen. Auf diese Weise lässt sich das Integral aus Gleichung 2.6 in eine Summe umformen. Für N Lichtquellen vereinfacht sich die Rendergleichung somit zu Gleichung 2.7 mit $\cos \theta \geq 0$.

vereinfachte Rendergleichung:

$$L_o(\mathbf{p}, \omega_o) = L_e(\mathbf{p}, \omega_o) + \sum_{k=1}^N f_r(\mathbf{p}, \omega_k, \omega_o) \frac{\Phi_k}{4\pi d_k^2} \cos \theta_k \quad (2.7)$$

$L_o(\mathbf{p}, \omega_o)$:	gesamte Strahldichte von \mathbf{p} in Richtung ω_o
$L_e(\mathbf{p}, \omega_o)$:	emittierte Strahldichte von \mathbf{p} in Richtung ω_o
$f_r(\mathbf{p}, \omega_k, \omega_o)$:	bidirektionale Reflektanzverteilungsfunktion
Φ_k :	Strahlungsleistung der Punktlichtquelle
θ_k :	Winkel zwischen ω_k und der Oberflächennormalen in \mathbf{p}
d_k :	Abstand zwischen \mathbf{p} und der Punktlichtquelle

Die Beleuchtung eines Punktes \mathbf{p} ist, für nicht emittierende Materialien in der vereinfachten Form, ausschließlich abhängig von der Betrachtungsrichtung ω_o , dem Abstand d_k sowie der Strahlungsleistung (Gleichung 2.5) aus N diskreten Richtungen ω_k . Dementsprechend eignet sich Gleichung 2.7 angesichts des deutlich niedrigeren Rechenaufwandes gut für das Rendern einer Szene in Echtzeit.

2.3 Dynamikkompression

Für das Rendern realitätsnaher Bilder ist es besonders wichtig mit radiometrischen Einheiten (Kapitel 2.1), oder zumindest radiometrischen Verhältnissen, zu arbeiten. Sämtliche Berechnungen sollten mit strahlungstechnischen Größen durchgeführt werden. Farben entstehen erst implizit bei der Visualisierung dieser Werte.

In der Displaytechnik werden üblicherweise normalisierte Bildsignale verarbeitet. Ein Signalwert $I_{in} = 0$ bildet den Schwarzpunkt, der Wert $I_{in} = 1$ den Weißpunkt. Die im Rendervorgang berechnete Strahldichte kann bei realitätsnahen Größen den Dynamikumfang eines Anzeigegerätes weitaus übertreffen. Um dennoch möglichst viele Bildinformationen zu erhalten, muss der Dynamikumfang des Bildes komprimiert werden.

Diese Dynamikkompression wird durch einen *Tone Mapping Operator* durchgeführt. Postprocessing-Effekte die auf linearer Mathematik basieren, wie beispielsweise Tiefen- und Bewegungsunschärfe, sollten daher unbedingt vor der Kompression angewendet werden (Abbildung 2.5).

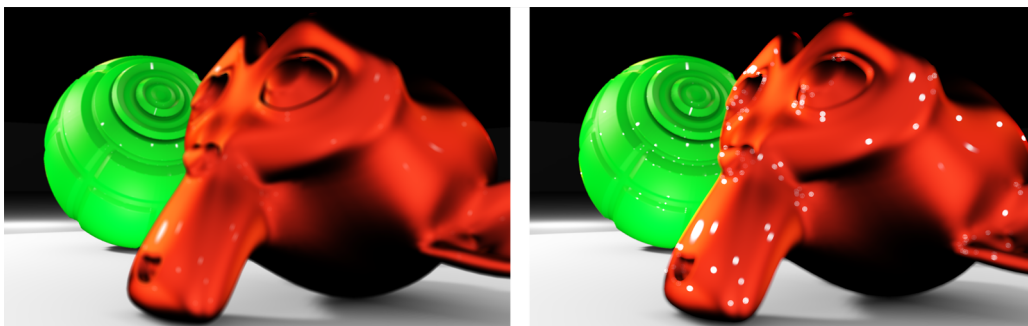


Abbildung 2.5 Ein Tiefenunschärfe-Effekt nach dem *Tone Mapping Operator* angewendet (links) und vorher (rechts). Wird der Effekt mit unkomprimierter Dynamik erzeugt, wirken sich Glanzlichtreflexionen korrekt auf die Unschärfe aus

Aus der Fotografie gibt es eine Vielzahl lokaler sowie auch globaler Operatoren. In der im Rahmen dieser Arbeit entwickelten Software wird der globale, exponentielle Tonemapping-Operator

$$L'(L, b) = 1 - e^{-bL} \quad (2.8)$$

mit $\lim_{L \rightarrow \infty} L'(L, b) = 1$ verwendet. In Abhängigkeit des Belichtungsfaktors b wird eine Strahldichte $\{L \in \mathbb{R} \mid 0 \leq L < \infty\}$ in eine von Anzeigegeräten darstellbare Größe $\{L' \in \mathbb{R} \mid 0 \leq L' < 1\}$ konvertiert (Abbildung 2.6).

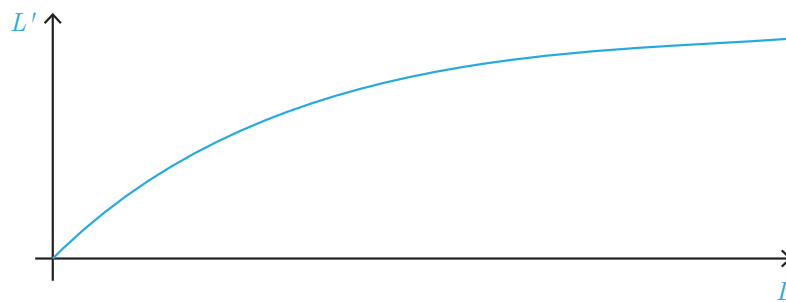


Abbildung 2.6 Der exponentielle Tone Mapping Operator mit $b = 1$

Da das visuelle System des Menschen wesentlich feinere Unterschiede in dunklen Bildbereichen wahrnimmt (Kapitel 2.4), bleiben so die wichtigsten Bildinformationen erhalten und ein Übersteuern des Ausgabesignals wird vermieden. Abbildung 2.7 zeigt die Auswirkung des Belichtungsfaktors.

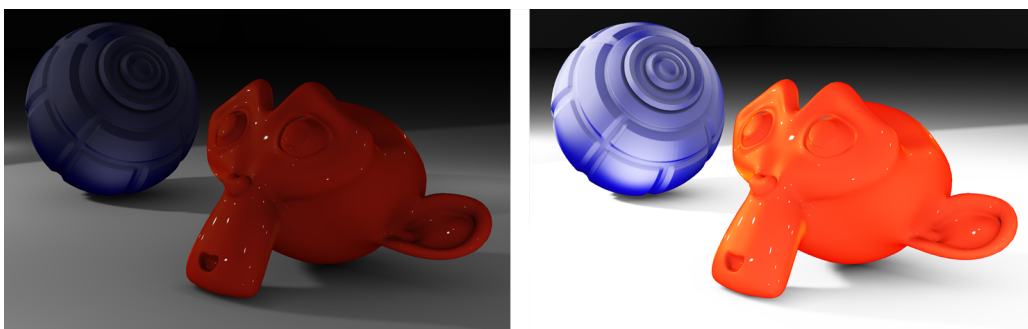


Abbildung 2.7 Unterbelichtung mit $b = 0.1$ (links) und Überbelichtung mit $b = 4.0$ (rechts)

2.4 Lineare Wiedergabe radiometrischer Größen

CRT-Monitore weisen einen nichtlinearen Zusammenhang zwischen der Spannung eines Eingangssignals und der resultierenden Strahldichte der Bildröhre auf [Ngu07]. Dieser Zusammenhang lässt sich mit der Exponentialfunktion $I_{out} = I_{in}^\gamma$ beschreiben.

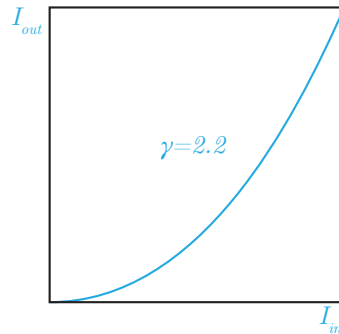


Abbildung 2.8 Display-Kennlinie mit $\gamma = 2.2$

In Abhängigkeit des γ -Wertes wird ein Eingangssignal I_{in} in ein Ausgangssignal I_{out} übertragen. Displays im Konsumerbereich weisen in der Regel einen γ -Wert von 2.2 auf (Abbildung 2.8). Die mittleren Signalamplituden werden gesenkt, Schwarz- und Weisspunkt bleiben erhalten. Aus Kompatibilitätsgründen besitzen auch digitale Displays eine solche Kennlinie.

Diese Kennlinie entspricht in etwa dem inversen menschlichen Hellempfinden [Ill04]. Das visuelle System des Menschen nimmt also einen linearen Strahldichteverlauf verzerrt wahr (Abbildung 2.9).

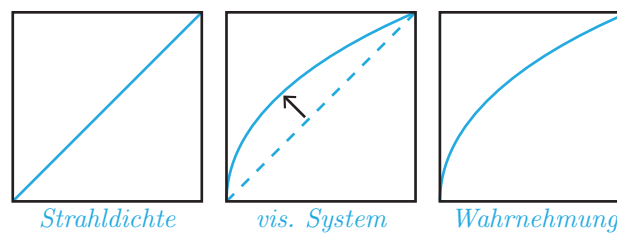


Abbildung 2.9 Die nichtlineare Wahrnehmung eines linearen Strahldichteverlaufs durch das visuelle System des Menschen

Für den realitätsnahen Eindruck einer gerenderten Szene ist es daher besonders wichtig, die ermittelte Strahldichte so wiederzugeben, dass die wahrgenommenen Helligkeitsverhältnisse denen einer realen Umgebung naheliegen.

Aufgrund der Display-Kennlinie in Verbindung mit dem visuellen System des Menschen, wird ein linearer Signalverlauf jedoch auch linear wahrgenommen (Abbildung 2.10) und entspricht somit nicht der Wahrnehmung eines realen, linearen Strahldichteverlaufs (Abbildung 2.9).

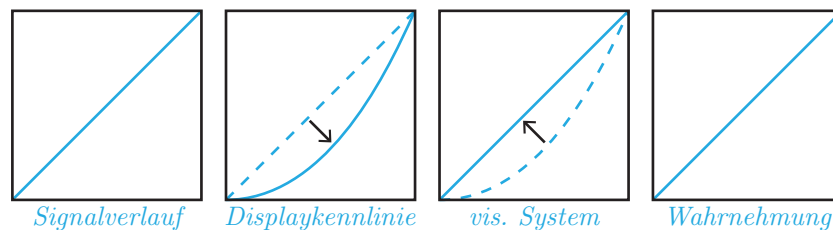


Abbildung 2.10 Die Übertragungsfunktion des Displays und das visuelle System des Menschen heben sich gegenseitig auf. Ein lineares Signal wird demnach linear wahrgenommen

Die Übertragungsfunktion des Displays muss daher mithilfe der Gamma-Vorentzerrung $I_{out} = I_{in}^{1/\gamma}$ kompensiert werden (Abbildung 2.11). Das lineare Signal wird hierfür mit dem Kehrwert des Display-Gamma-Wertes vorentzerzt.

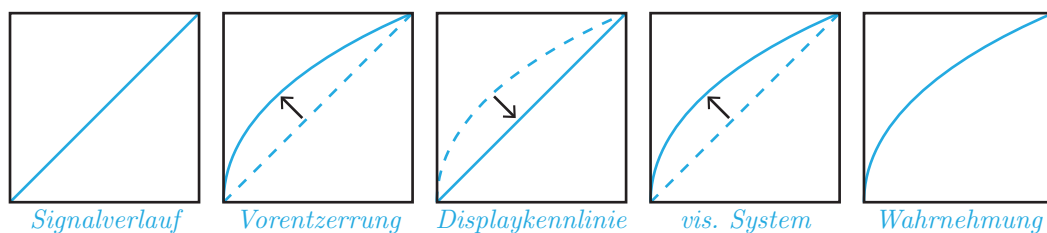


Abbildung 2.11 Ein linearer Signalverlauf wird vorentzerzt um die Ausgabe eines linearen Strahldichteverlaufs durch das Wiedergabegerät zu bewirken. Hierdurch entspricht die Wahrnehmung der einer realen Szene (Abbildung 2.9)

Somit wird die Verzerrung des Monitors im Voraus aufgehoben und das Bild wird linear wiedergegeben.

In der Videotechnik passiert die Vorentzerrung des Signals schon direkt in der Kamera bei der Bildaufnahme, in der Computergrafik sollte die Vorentzerrung als letzter Schritt vor der Bildwiedergabe passieren.

Wird ein Signal nicht linear wiedergegeben, kommt es zu Farbtonveränderungen und unnatürlich weichem Helligkeitsabfall [Ngu07]. Abbildung 2.12 zeigt den Effekt der Gamma-Vorentzerrung auf den Helligkeitsabfall.

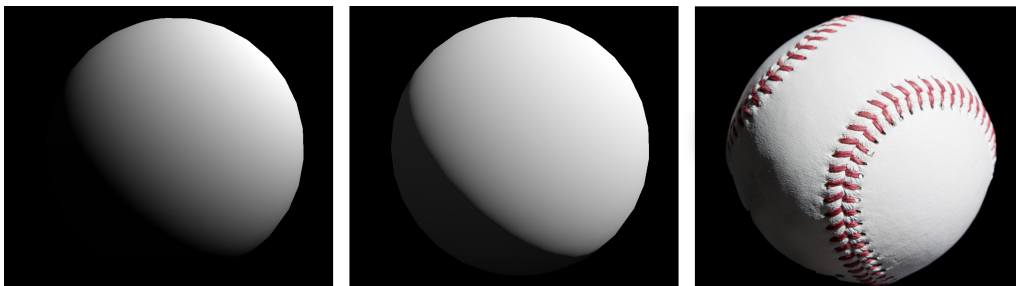


Abbildung 2.12 *Ein durch die Kennlinie des Monitors verzerrtes Signal (links), lineare Ausgabe durch Vorentzerrung (Mitte) und ein Vergleichsfoto (rechts)*

3 Lokale Beleuchtungsmodelle

Lokale Beleuchtungsmodelle beschreiben das Reflexionsverhalten eines Punktes auf der Oberfläche der Geometrie einer Szene. Dieses Reflexionsverhalten wird in der modernen Rendergleichung (Gleichung 2.6) in Form einer *bidirektionalen Reflektanzverteilungsfunktion* (engl. *Bidirectional Reflectance Distribution Function*, *BRDF*) dargestellt.

3.1 Die bidirektionale Reflektanzverteilungsfunktion

Die bidirektionale Reflektanzverteilungsfunktion wurde 1965 zuerst von Fred Nicodemus definiert [Nic65]. In der modernen Form entspricht die BRDF $f_r(\mathbf{p}, \omega_i, \omega_o)$ in einem Punkt \mathbf{p} dem Verhältnis der Strahldichte L_o zu der Bestrahlungsstärke E [AMHH08]:

$$f_r(\mathbf{p}, \omega_i, \omega_o) = \frac{dL_o(\mathbf{p}, \omega_o)}{dE(\mathbf{p}, \omega_i)} [\text{sr}^{-1}] \quad (3.1)$$

Eine BRDF ist ausschließlich abhängig von dem Punkt \mathbf{p} , der Betrachtungsrichtung ω_o sowie der Richtung des einfallenden Lichts ω_i (Abbildung 3.1).

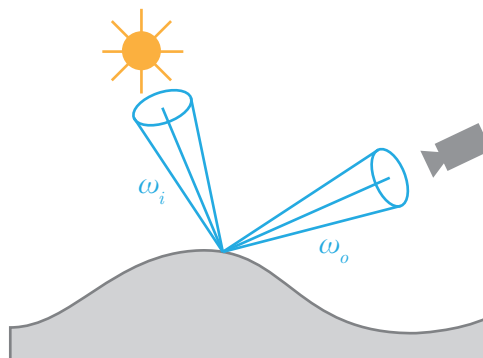


Abbildung 3.1 Veranschaulichung der Richtungsabhängigkeit einer BRDF

Um die physikalische Plausibilität einer BRDF zu gewährleisten, müssen zwei Randbedingungen erfüllt sein [PH10]. Die erste Voraussetzung ist die Einhaltung der Helmholtz-Reziprozität:

$$f_r(\mathbf{p}, \omega_i, \omega_o) = f_r(\mathbf{p}, \omega_o, \omega_i) \quad (3.2)$$

Eine BRDF liefert somit das gleiche Ergebnis wenn die Richtungen ω_i und ω_o ausgetauscht werden, es wird also die gleiche Strahldichte in Richtung des Betrachters reflektiert wenn die Positionen von Lichtquelle und Betrachter vertauscht werden.

Als zweite Randbedingung steht die Energieerhaltung. Aufgrund des Energieerhaltungssatzes gilt

$$\forall \omega_i, 0 \leq \int_{\Omega} f_r(\mathbf{p}, \omega_i, \omega_o) \cos \theta_i d\omega_i \leq 1. \quad (3.3)$$

Die von einem Punkt reflektierte Energie ist somit stets positiv und nie größer als die eintreffende Energie.

Werden diese beiden Voraussetzungen erfüllt, wird die BRDF *physikalisch plausibel* genannt.



Abbildung 3.2 *v.l.n.r.: Oberflächen steigender Rauheit gerendert mit einer physikalisch plausiblen BRDF*

Abbildung 3.2 zeigt die Energieerhaltung anhand einer Reihe verschieden rauer Oberflächen. Zu beachten ist hier die Helligkeit der Glanzreflexion in Abhängigkeit der Oberflächenrauheit. Mit steigender Rauheit wird das Licht stärker gestreut, Glanzreflexionen erscheinen größer und weniger hell.

Bei den verschiedenen BRDFs unterscheidet man zwischen isotropen und anisotropen Funktionen. Ändert sich die Form der Glanzlichtreflexion nicht mit der Rotation einer Oberfläche um die Oberflächennormale, wird die bidirektionale Reflektanzverteilungsfunktion *isotrop* genannt.

Ändert sich aber die Form der Glanzlichtreflexion, wird die BRDF *anisotrop* genannt. Beispiele für anisotrope Materialien sind Haare, gebürstetes Metall (Abbildung 3.3) oder Holz.



Abbildung 3.3 Gebürstetes Metall dargestellt mit einer anisotropen BRDF, gerendert in RenderMonkey

In der Realität entsteht ein solches Reflexionsverhalten durch feine, richtungsabhängige Unebenheiten einer Oberfläche. Diese Unebenheiten können beispielsweise Materialfasern bei Stoffen sein, oder durch den Herstellungsprozess einer Sache, anhand von Arbeitsschritten wie Schleifen oder Fräsen, entstehen.

Für die Darstellung anisotroper Oberflächen muss nicht nur die Oberflächennormale, sondern der Tangentialraum in einem Punkt bekannt sein. Hierfür kann der gleiche Tangentialraum wie auch für das Normal Mapping verwendet werden, sofern dieser kontinuierlich ist. Ist dies nicht der Fall, müssen separate, kontinuierliche Tangenteninformationen in den Vertexattributen oder in Texturen gespeichert werden, da sonst sichtbare Artefakte in der Glanzreflexion anisotroper BRDFs an den Diskontinuitäten des Tangentialraums auftreten.

Das Reflexionsverhalten eines Materials kann nicht immer mit einer einzigen BRDF-Form modelliert werden. Für die Darstellung von komplexeren Oberflächenstrukturen kann die gewichtete Summe verschiedener BRDF-Formen verwendet werden. Für $\sum_{k=1}^N c_k = 1$ gilt

$$f_r(\mathbf{p}, \omega_i, \omega_o) = \sum_{k=1}^N c_k f_{rk}(\mathbf{p}, \omega_i, \omega_o). \quad (3.4)$$

In Abbildung 3.4 ist die Kombination von zwei unterschiedlichen metallischen Oberflächenstrukturen zu einem komplexeren Material dargestellt.

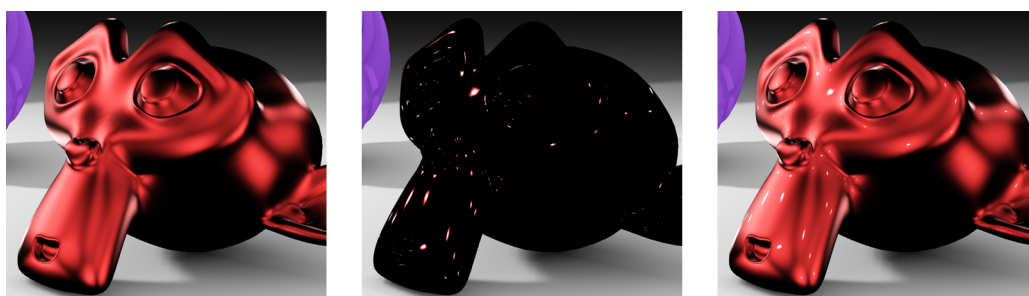


Abbildung 3.4 *Kombination eines rauen Metalls (links) mit einer glatten Metalloberfläche (Mitte) zu einem Material gemischter Oberflächenstruktur (rechts)*

Generell wird in der Computergrafik zwischen gerichteter und diffuser Reflexion unterschieden (Abbildung 3.5). Die Gewichtung dieser Anteile sollte durch den Reflexionsgrad des Materials erfolgen (Kapitel 3.2).



Abbildung 3.5 *Komposition (rechts) eines Materials aus Diffus- (links) und Glanzanteil (Mitte) nach Gleichung 3.4*

3.2 Der Brechungsindex

Der Brechungsindex n_M ist eine dimensionslose, wellenlängenabhängige Materialkonstante, die beschreibt wie schnell sich Licht im Vakuum in Relation zur Lichtgeschwindigkeit in dem Medium c_M ausbreitet:

$$n_M = \frac{c_0}{c_M} \quad (3.5)$$

Für lichtundurchlässige Medien, in der Regel Metalle, ist der Brechungsindex eine komplexe Zahl. Der Imaginärteil einer komplexen Brechzahl ist verantwortlich für den Absorptionsfaktor des Materials.

Trifft ein Lichtstrahl auf eine Grenzfläche zwischen zwei lichtdurchlässigen Medien unterschiedlicher Brechzahl, so wird der Strahl nach dem Snellius-Gesetz gebrochen (Abbildung 3.6). In der Echtzeit-Computergrafik wird dieser Effekt meist gänzlich ausgelassen.

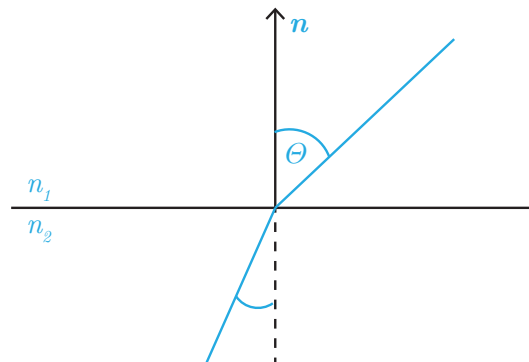


Abbildung 3.6 Brechung eines Lichtstrahls an der Grenzfläche zwischen Medien unterschiedlicher Brechzahl

Dennoch ist der Brechungsindex ein erheblicher Bestandteil realitätsnaher Bildsynthese. Mithilfe der fresnelschen Formeln lässt sich anhand der Brechzahl das Verhältnis von Reflexion zu Transmission glatter Oberflächen bestimmen. Diese Relation ist nicht konstant, sondern variiert mit dem Betrachtungswinkel (Abbildung 3.7).

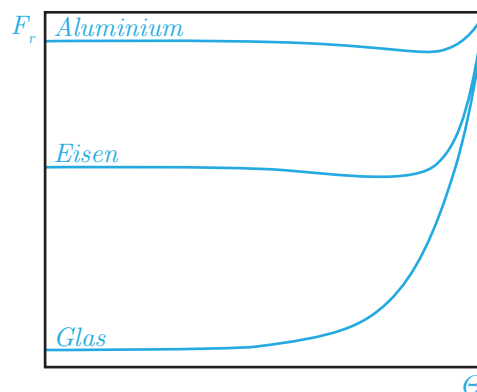


Abbildung 3.7 Reflexionsgrade verschiedener Brechungsindizes in Relation zu dem Betrachtungswinkel

Um die exakten Reflexions- und Transmissionsfaktoren zu bestimmen, müssen die fresnelschen Gleichungen in Abhängigkeit der Wellenlänge, sowie Real- und Imaginärteil des Brechungsindex, gelöst werden. Dieser Aufwand erweist sich als nicht praktikabel in Echtzeitanwendungen.

Die in der Computergrafik geläufigste Näherung der fresnelschen Gleichungen wurde 1994 von Christophe Schlick veröffentlicht [Sch94]:

$$F_r(\theta) = f_{\perp} + (1 - f_{\perp})(1 - \cos \theta)^5 \quad (3.6)$$

Der Reflexionsgrad F_r wird in Abhängigkeit des Betrachtungswinkels θ , sowie des Reflexionsgrades bei senkrechter Betrachtung $f_{\perp} = F_r(0^\circ) = \left(\frac{n_1 - n_2}{n_1 + n_2}\right)^2$ beschrieben. Schlicks Näherung erfüllt folgende Kernbedingungen des exakten Reflexionsgrades:

- Die Funktion $F_r(\theta)$ ist bei flachen Betrachtungswinkeln gleich 1.
- Die Steigung der Funktion ist bei senkrechter Betrachtung gleich 0.
- Die Krümmung der Funktion ist bei senkrechter Betrachtung gleich 0.

Laut Schlick liegt der Fehler der Näherung bei unter einem Prozent, es wird also eine durchaus akzeptable Approximation der exakten Werte mit geringem Aufwand erreicht.

Für dielektrische Materialien ist der Reflexionsfaktor, bei Vernachlässigung der Wellenlängenabhängigkeit, ein einfacher Skalar.

Der Farbeindruck metallischer Oberflächen entsteht jedoch ausschließlich durch den komplexen, wellenlängenabhängigen Brechungsindex des Metalls (Abbildung 3.8). Aufgrund der Breite des Spektrums und der erforderlichen Farbkonversion eignet sich die Berechnung von $f_{\perp}(\lambda)$ nicht für Echtzeitanwendungen.



Abbildung 3.8 Der wellenlängenabhängige Reflexionsgrad von Metallen (links: Kupfer, rechts: Gold) ist maßgebend für den Farbeindruck des Materials

Es bietet sich jedoch an diesen Wert vorzuberechnen. Für die Darstellung von Metallen wurde in der im Rahmen dieser Arbeit entwickelten Software die Tabelle vorberechneter Reflexionsgrade aus [AMHH08] verwendet. Das Spektrum dieser Werte wurde anhand der CIE-Spektralwertfunktionen in RGB-Farbvektoren konvertiert. Dies ermöglicht eine komponentenweise Ermittlung des Reflexionsgrades pro Farbkanal und eignet sich daher gut für das Echtzeitrendering realitätsnaher Metalle, modelliert aber nicht die Farbveränderung bei variierendem Betrachtungswinkel.

Um diesen Effekt zu modellieren, kann eine eindimensionale Wertetabelle in Form einer Textur verwendet werden. So ist es möglich $F_r(\theta)$ direkt ohne Berechnung anhand der Wertetabelle zu evaluieren.

3.3 Gerichtete Reflexion

Die gerichtete Reflexion ist eine Form der Spiegelung. Je glatter eine Oberfläche, desto gerichteter wird das Licht auf dieser Oberfläche reflektiert. Zur Berechnung des Reflexionsverhaltens einer Oberfläche existieren eine Vielzahl lokaler Beleuchtungsmodelle.

3.3.1 Das Phong-Modell

Das Phong-Beleuchtungsmodell wurde 1975 von Bui Tuong Phong entwickelt [Pho75] und ist bis heute aufgrund des geringen Rechenaufwandes eines der wichtigsten Beleuchtungsmodelle in der Computergrafik. Die moderne Form lautet

$$I_o = I_a + I_d + I_s. \quad (3.7)$$

Die Lichtintensität in einem Punkt ist nach Phong die Summe des ambienten, diffusen und gerichteten Lichts. Eine Besonderheit des Modells zum Zeitpunkt der Veröffentlichung war die, im Vergleich zu zeitgemäßen Methoden wie dem Gouraud Shading [Gou71], deutlich höhere Qualität der Glanzreflexionen. Diese Qualität wurde durch die korrekte Interpolation der Oberflächennormalen eines Polygons, gekoppelt mit der Lichtberechnung pro Bildpunkt, erreicht.

Im weiteren Verlauf dieses Kapitels wird ausschließlich die gerichtete Reflexion des Phong-Modells betrachtet.

Die BRDF-Form der Phong-Glanzreflexion ergibt sich aus einer wellenlängen-abhängigen Konstante $\{k_s \in \mathbb{R} \mid 0 \leq k_s \leq 1\}$ sowie der n -ten Potenz des Kosinus des Winkels α zwischen Betrachter und idealer Reflexion (Abbildung 3.9):

$$f_{Phong} = k_s \cos(\alpha)^n. \quad (3.8)$$

Je größer der Parameter n , desto gerichteter ist die Reflexion.

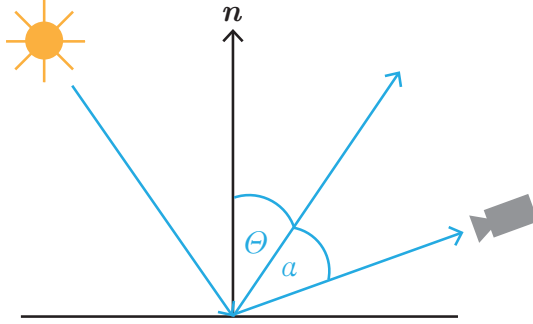


Abbildung 3.9 Veranschaulichung der geometrischen Zusammenhänge von Gleichung 3.8

Das Phong-Modell ist ein rein phänomenologisches Beleuchtungsmodell. Es basiert nicht auf Grundlagen der Physik oder Optik, sondern versucht lediglich den Eindruck realistischer Beleuchtung zu erzeugen.

In der Standardform verletzt das Phong-Modell den Energieerhaltungssatz. Je nach Parameterwahl kann die reflektierte Energie deutlich größer sein als die eintreffende Energie. Für eine physikalisch plausible Beleuchtung muss Gleichung 3.8 normalisiert werden [LW94]. Durch das Normalisieren einer BRDF wird sichergestellt, dass der Energieerhaltungssatz eingehalten wird. Hierfür muss der Normalisierungsfaktor der BRDF, also der Kehrwert des Maximums $\rho_{PhongMax}$, gefunden werden.

Die maximale Reflexion tritt für das Phong-Modell ein, wenn die Oberflächennormale auf die Lichtquelle zeigt. Für diesen Fall entspricht der Reflexionsvektor der Oberflächennormalen, somit ist $\alpha = \theta$. Mit $k_s = 1$ ergibt sich der Maximalwert der Phong-Glanzreflexion durch

$$\begin{aligned}
 \rho_{PhongMax} &= \int_{\Omega} \cos(\theta)^n \cos \theta d\omega_i \\
 &= \int_{\Omega} \cos(\theta)^{n+1} d\omega_i \\
 &= \int_0^{2\pi} \int_0^{\pi/2} \cos(\theta)^{n+1} \sin \theta d\theta d\phi \\
 &= \frac{2\pi}{n+2}.
 \end{aligned} \tag{3.9}$$

Ersetzt man k_s in Gleichung 3.8 mit dem Normalisierungsfaktor $\frac{n+2}{2\pi}$, so erhält man die physikalisch plausible Phong-BRDF

$$f_{Phong} = \frac{n+2}{2\pi} \cos(\alpha)^n. \quad (3.10)$$

Neben dem geringen notwendigen Rechenaufwand bietet das Phong-Reflexionsmodell einen weiteren Vorteil: Durch die rotationssymmetrische BRDF-Form (Abbildung 3.10) lassen sich Umgebungstexturen vorfiltern [KM00]. Dies ermöglicht die effiziente Darstellung gestreuter Glanzreflexionen, indem zur Laufzeit lediglich eine gefilterte Umgebungstextur evaluiert wird.

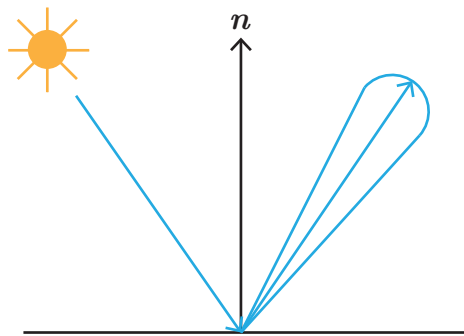


Abbildung 3.10 Darstellung des rotationssymmetrischen Reflexionskegels des Phong-Beleuchtungsmodells

Auch lässt sich die Beleuchtung einer Oberfläche durch polygonale Flächenstrahler analytisch für die Phong-BRDF ermitteln [Sny96].

Es ist jedoch selbst mit dem physikalisch plausiblen Phong-Modell nicht möglich, einen Großteil vermessener Oberflächen mit ausreichender Präzision darzustellen.

1977 wurde das Phong-Modell von James Blinn neu parametrisiert [Bli77]. Im Gegensatz zum Phong-Modell basiert Blinns Methode auf physikalischen Erkenntnissen von Torrance und Sparrow (Kapitel 3.3.2).

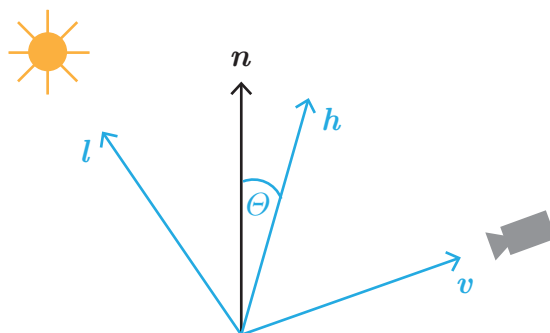


Abbildung 3.11 Darstellung der Parameter des Blinn-Phong-Modells

Durch die Verwendung des Winkels θ zwischen der Oberflächennormalen \mathbf{n} und der Winkelhalbierenden \mathbf{h} zwischen Licht- und Betrachtungsrichtung (Abbildung 3.11) entspricht die Form der Glanzlichtreflexion, besonders bei flacher Betrachtungsrichtung, deutlich stärker den auf realen Oberflächen auftretenden Reflexionen [NDM04] (Abbildung 3.12).

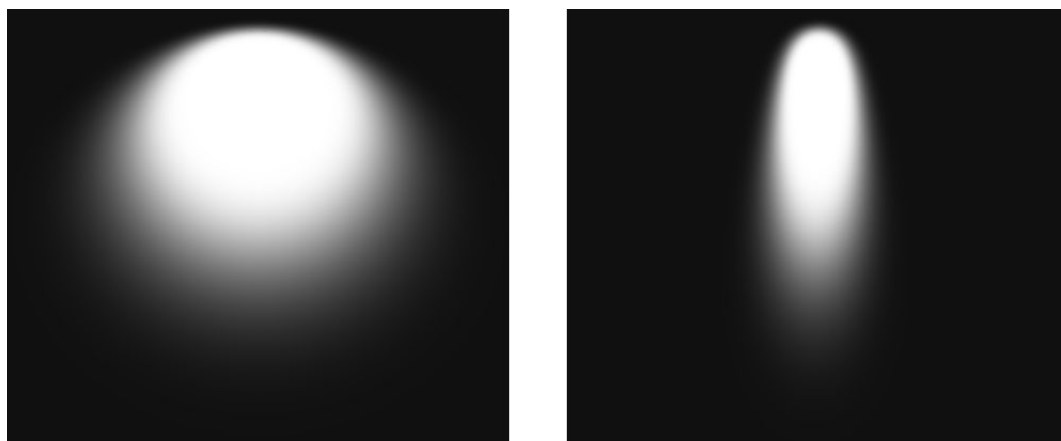


Abbildung 3.12 Vergleich zwischen Phong- (links) und Blinn-Phong-Glanzreflexionen (rechts)

Die Winkelhalbierende \mathbf{h} ist für die normalisierten Vektoren \mathbf{v} und \mathbf{l} definiert als

$$\mathbf{h} = \frac{\mathbf{v} + \mathbf{l}}{\|\mathbf{v} + \mathbf{l}\|}. \quad (3.11)$$

Somit ergibt sich die Blinn-Phong-BRDF mit $\cos \theta = \mathbf{n} \cdot \mathbf{h}$ zu

$$f_{Blinn} = k_s (\mathbf{n} \cdot \mathbf{h})^n. \quad (3.12)$$

Wie auch die ursprüngliche Phong-BRDF, vernachlässigt Blinns Version die Energieerhaltung. Aufgrund der Komplexität des exakten Normalisierungsfaktors kann $\frac{n+8}{8\pi}$ als Näherung verwendet werden [AMHH08].

Die normalisierte Blinn-Phong-BRDF ist demnach

$$f_{Blinn} = \frac{n+8}{8\pi} (\mathbf{n} \cdot \mathbf{h})^n. \quad (3.13)$$

Das normalisierte Blinn-Phong-Modell ist aufgrund des geringen Verhältnisses von Rechenaufwand zu der darstellbaren Materialvielfalt ein sehr beliebtes Beleuchtungsmodell in der Videospielindustrie.

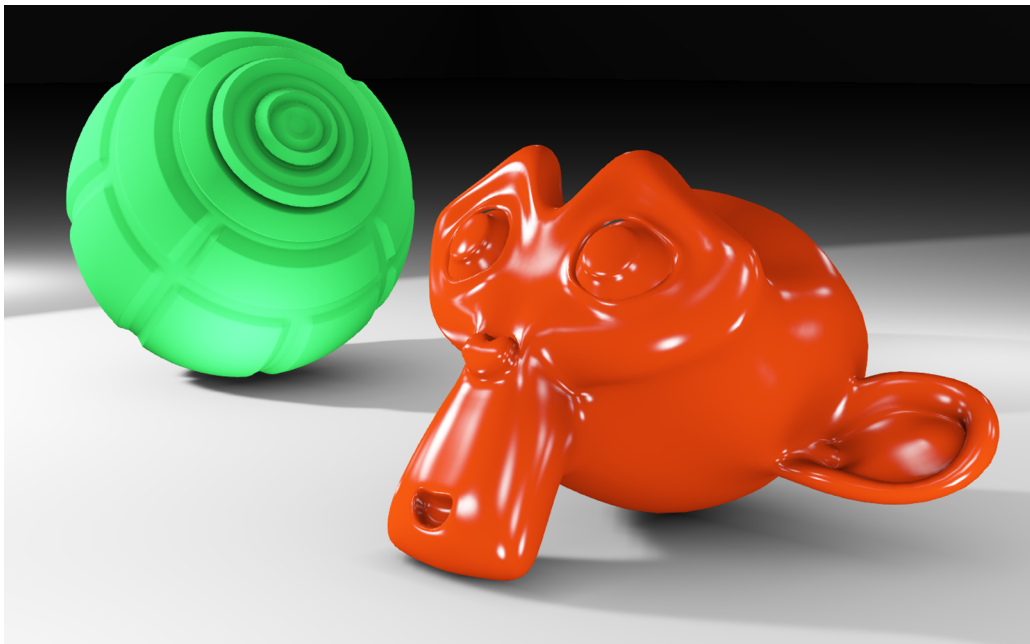


Abbildung 3.13 *Blinn-Phong-BRDF Glanzlichtreflexionen*

3.3.2 Das Torrance-Sparrow-Modell

Ursprünglich wurde das Torrance-Sparrow-Modell 1967 als physikalisches Modell zur Beschreibung des Reflexionsverhaltens rauer Oberflächen eingeführt [TS67]. Fünf Jahre später von Cook und Torrance [CT82] für die Computergrafik adaptiert, bietet das Torrance-Sparrow-Modell neben dem Ashikhmin-Shirley-Modell noch bis heute die Repräsentationen vermessener Materialien mit der geringsten Fehlerabweichung [NDM04].

Das Torrance-Sparrow-Modell beruht auf der Mikrofacettentheorie. Es wird angenommen, eine raue Oberfläche bestehe aus einer Vielzahl verschieden ausgerichteter, ideal reflektierender Spiegel.

Die Cook-Torrance-BRDF ist

$$f_{CT}(\mathbf{p}, \mathbf{l}, \mathbf{v}) = \frac{F_r D G}{\pi (\mathbf{n} \cdot \mathbf{l})(\mathbf{n} \cdot \mathbf{v})}. \quad (3.14)$$

F_r ist der Reflexionsgrad wie in Kapitel 3.2 beschrieben, D das Verhältnis perfekt spiegelnd ausgerichteter Mikrofacetten (Abbildung 3.14) und G modelliert Abschattung und Maskierung (Abbildung 3.15) durch die Geometrie der Mikrofacetten selbst. Interreflexionen werden vernachlässigt.

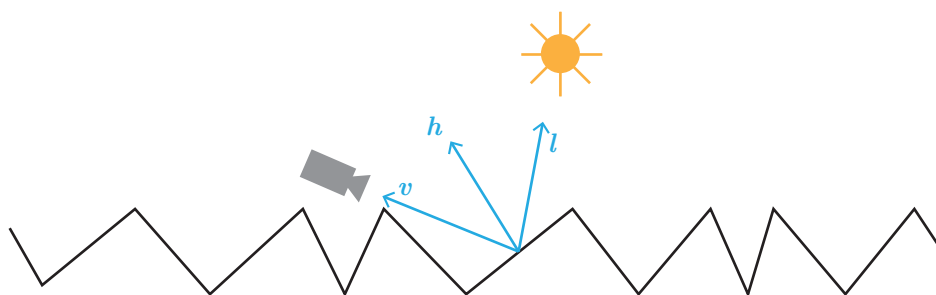


Abbildung 3.14 Die Spiegelung an einer Mikrofacette ist gegeben, wenn die Oberflächennormale der Mikrofacette dem winkelhalbierenden Vektor \mathbf{h} , zwischen Betrachtungsrichtung \mathbf{v} und Licht-richtung \mathbf{l} , entspricht.

Die Komponenten der BRDF sind modular, es können verschiedene Funktionen zur Beschreibung der einzelnen Effekte verwendet werden, sofern die Randbedingungen physikalisch plausibler BRDFs nicht verletzt werden (Kapitel 3.1).

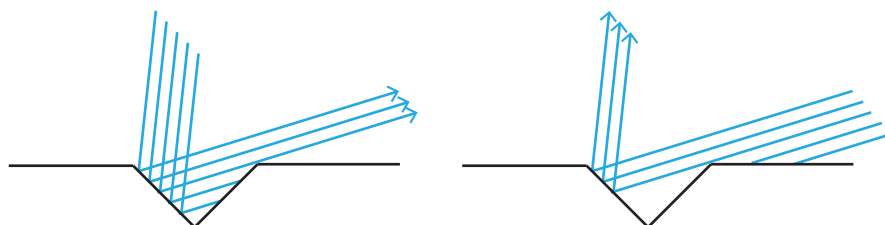


Abbildung 3.15 *Durch die geometrische Anordnung der Mikrofacetten wird ein Teil des reflektierten Lichts maskiert (links) oder schattiert (rechts).*

Die Beckmann-Verteilungsfunktion ist in der Lage eine große Vielfalt unterschiedlicher Materialien darzustellen. Die Funktion lautet

$$D_{Beck} = \frac{1}{m^2 \cos^4 \alpha} e^{-\left(\frac{\tan \alpha}{m}\right)^2} \quad (3.15)$$

mit $\alpha = \arccos(\mathbf{n} \cdot \mathbf{h})$. Der Parameter m ist der quadratische Mittelwert der Steigung der Mikrofacetten.

Eine äquivalente, für Computer effizienter zu berechnende Form ist

$$D_{Beck} = \frac{1}{m^2 \cos^4 \alpha} e^{\frac{\cos^2(\alpha) - 1}{\cos^2(\alpha) m^2}}. \quad (3.16)$$

Der Abschattungs- und Maskierungsfaktor G ist nach [CT82]

$$G = \min\left\{1, \frac{2(\mathbf{n} \cdot \mathbf{h})(\mathbf{n} \cdot \mathbf{v})}{(\mathbf{v} \cdot \mathbf{h})}, \frac{2(\mathbf{n} \cdot \mathbf{h})(\mathbf{n} \cdot \mathbf{l})}{(\mathbf{v} \cdot \mathbf{h})}\right\}. \quad (3.17)$$

Eine Oberfläche kann also mit der Cook-Torrance-BRDF durch lediglich zwei Parameter beschrieben werden, dem Brechungsindex des Materials n sowie dem quadratischen Mittelwert der Steigung der Mikrofacetten m .

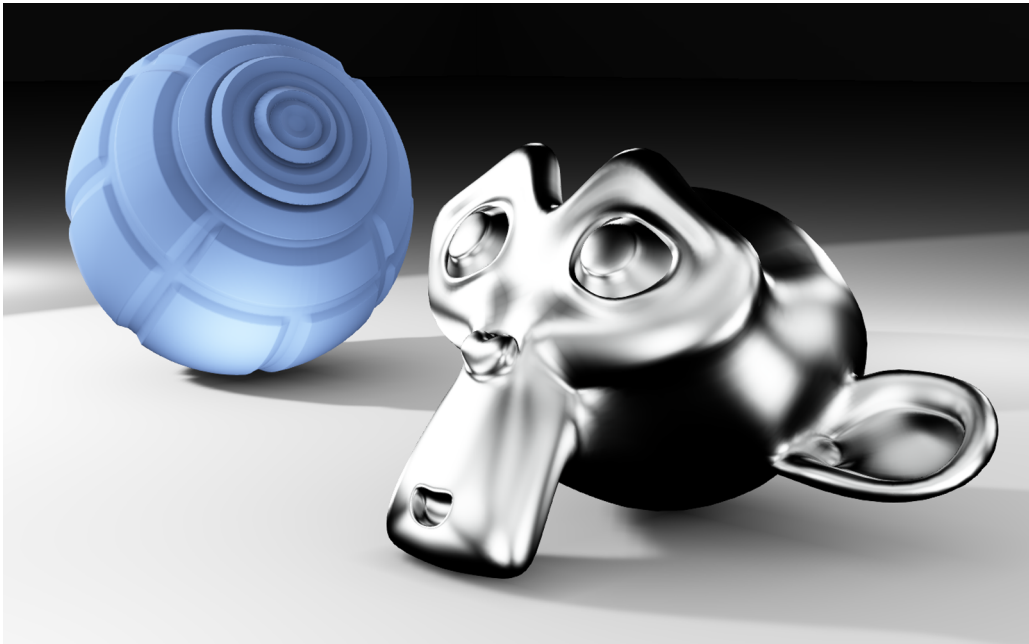


Abbildung 3.16 *Aluminium*



Abbildung 3.17 *Kunststoff*



Abbildung 3.18 *Metalleffektlack*

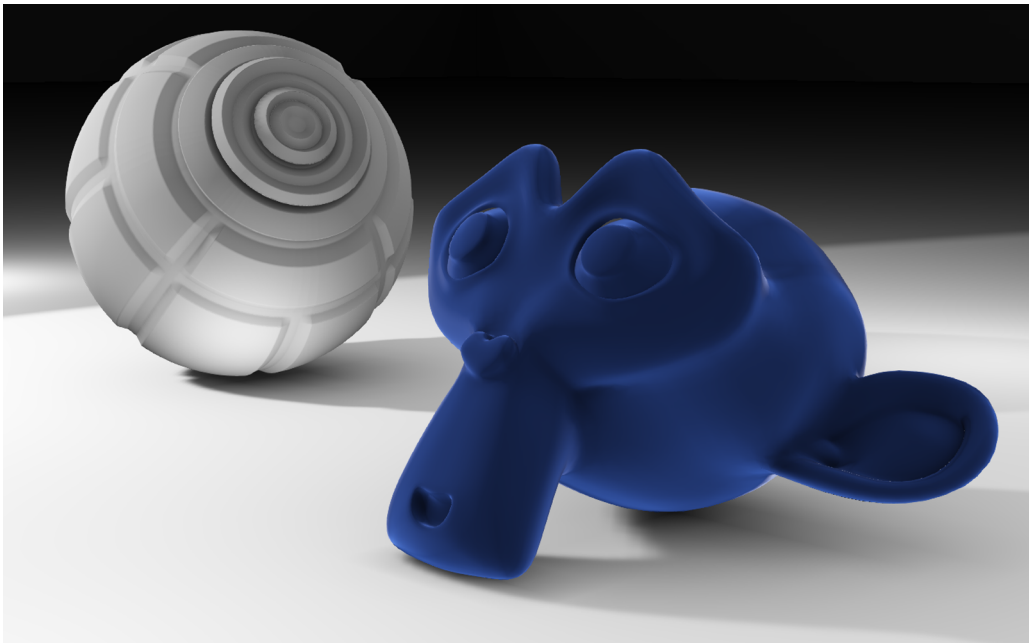


Abbildung 3.19 *stumpfe Metalloberfläche*

3.4 Diffuse Reflexion

Die ideal diffuse Reflexion streut Licht in alle Richtungen gleichermaßen und tritt bei dielektrischen Materialien auf. Die Ursache ist die Streuung des Lichts unter der Oberfläche des Materials [AMHH08].

Dielektrische Oberflächen lösen einen Farbreiz aus, wenn ein Teil des Spektrums, von beispielsweise Farbpigmenten, absorbiert wird (Abbildung 3.20).

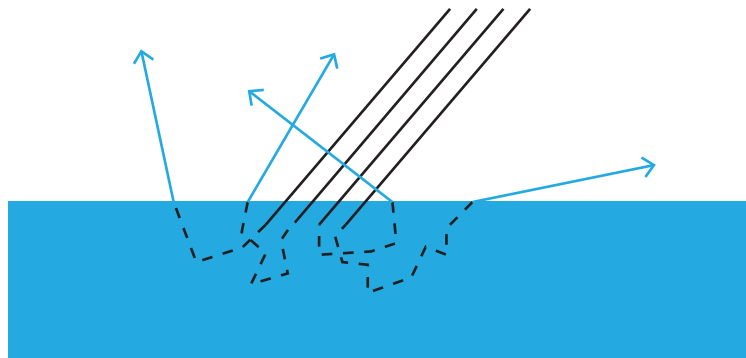


Abbildung 3.20 Die Lichtstreuung und Absorption in den Farbpigmenten eines Materials ist verantwortlich für den diffusen Farbeindruck einer Oberfläche

In der Realität besitzt nahezu jede Reflexion einer dielektrischen Oberfläche sowohl diffuse als auch gerichtete Anteile.

3.4.1 Der Lambertsche Reflektor

Eine ideal diffus reflektierende Fläche wird auch *Lambertscher Reflektor* genannt. Als BRDF ausgedrückt ist ein Lambertscher Reflektor eine einfache Konstante:

$$f_{diffus}(\omega_i, \omega_o) = \rho \quad (3.18)$$

Der Albedo-Wert $\{\rho \in \mathbb{R} \mid 0 \leq \rho \leq 1\}$ ist eine spektrale Größe und bestimmt den durch Betrachtung des Materials wahrgenommenen Farbreiz.

Gleichung 3.18 ist allerdings nicht energieerhaltend. Der Normalisierungsfaktor der BRDF ergibt sich, wie auch in Kapitel 3.3.1 beschrieben, aus dem Kehrwert des maximalen Reflexionsfaktors ρ_{max} mit $\rho = 1$:

$$\begin{aligned}\rho_{max} &= \int_{\Omega} \cos \theta d\omega_i \\ &= \int_0^{2\pi} \int_0^{\pi/2} \cos \theta \sin \theta d\theta d\phi \\ &= \pi\end{aligned}\tag{3.19}$$

Demnach ist die physikalisch plausible BRDF eines ideal diffusen Reflektors

$$f_{diffus}(\omega_i, \omega_o) = \frac{\rho}{\pi}.\tag{3.20}$$

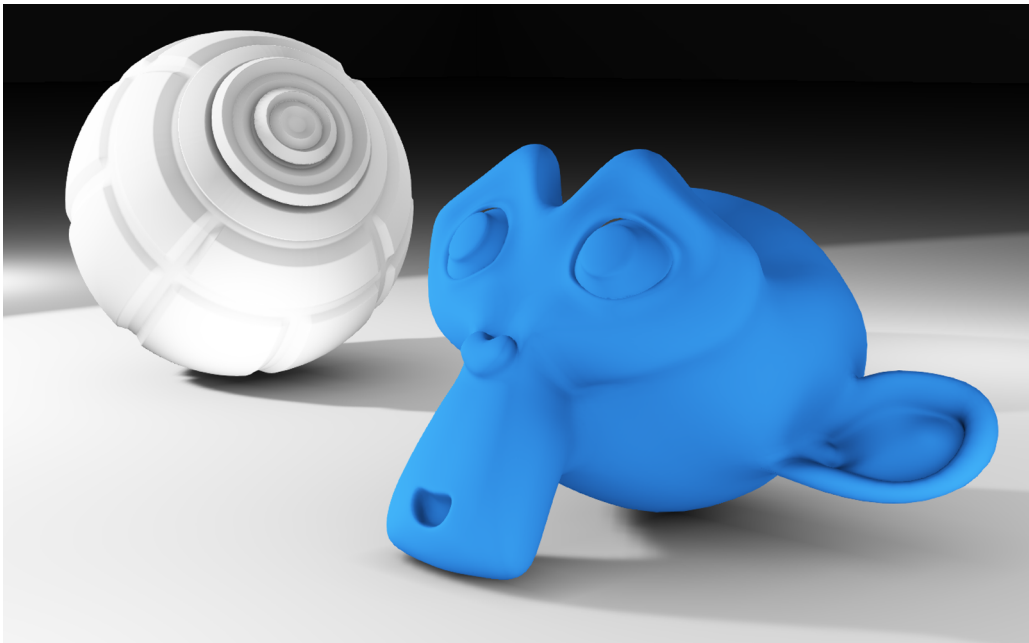


Abbildung 3.21 *Ideal diffus reflektierende Oberflächen nach Lambert*

3.4.2 Das Oren-Nayar-Modell

In vielen Fällen lässt sich eine diffus reflektierende Oberfläche aufgrund feiner Unebenheiten nicht anhand des Lambertschen Reflektors darstellen.

Die Oren-Nayar-BRDF modelliert diese Unebenheiten, ähnlich dem Torrance-Sparrow-Modell (Kapitel 3.3.2), durch Mikrofacetten [ON94]. Diese Mikrofacetten stellen jedoch nicht, wie bei der gerichteten Reflexion, ideale Spiegel dar, sondern ideal diffuse Facetten.

Zusätzlich zu Maskierung und Abschattung modelliert das Oren-Nayar-Modell auch die Interreflexionen zwischen den Mikrofacetten (Abbildung 3.22).

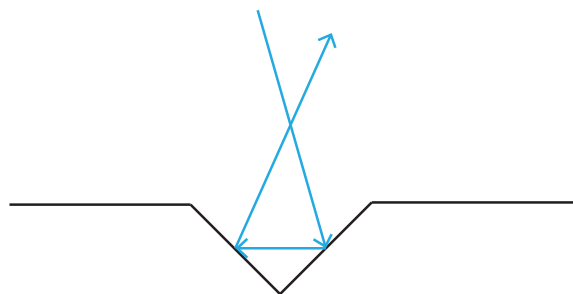


Abbildung 3.22 Die Interreflexion zwischen Mikrofacetten

Nach [AMHH08] lautet die vereinfachte Oren-Nayar-BRDF

$$f_{ON}(\omega_i, \omega_o) = \frac{\rho}{\pi} (A + B \cos \phi \sin(\min\{\theta_i, \theta_o\}) \tan(\max\{\theta_i, \theta_o\})). \quad (3.21)$$

mit

$$\begin{aligned} A &= 1 - 0.5 \frac{\sigma^2}{\sigma^2 + 0.33} \\ B &= 0.45 \frac{\sigma^2}{\sigma^2 + 0.09}. \end{aligned} \quad (3.22)$$

Das Maß der Oberflächenrauheit $\{\sigma \in \mathbb{R} \mid 0 \leq \sigma \leq \frac{\pi}{2}\}$ ist als die Standardabweichung des Winkels zwischen der makroskopischen Oberflächennormalen und der Normalen der Mikrofacetten definiert. Abbildung 3.23 zeigt den geometrischen Aufbau zu Gleichung 3.21.

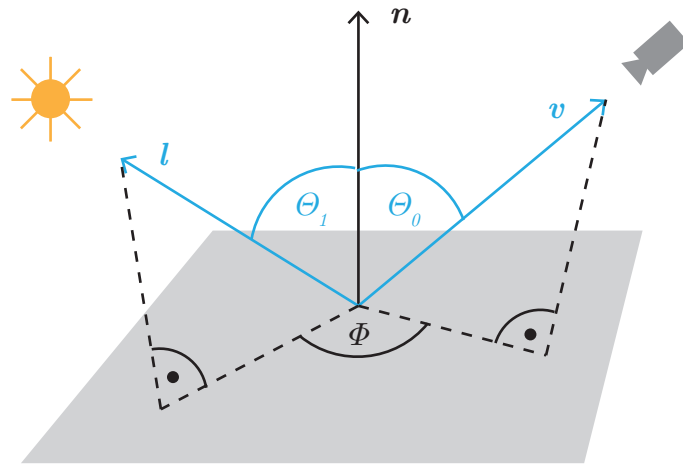


Abbildung 3.23 Die Winkel und Richtungen der Oren-Nayar-BRDF

Das Oren-Nayar-Modell wird aufgrund des benötigten Rechenaufwandes überwiegend im Offline-Rendering verwendet. Auf aktuell verfügbaren Grafikkarten ist das Modell durch die ausreichende Grafikleistung jedoch auch für das Echtzeitrendering geeignet.

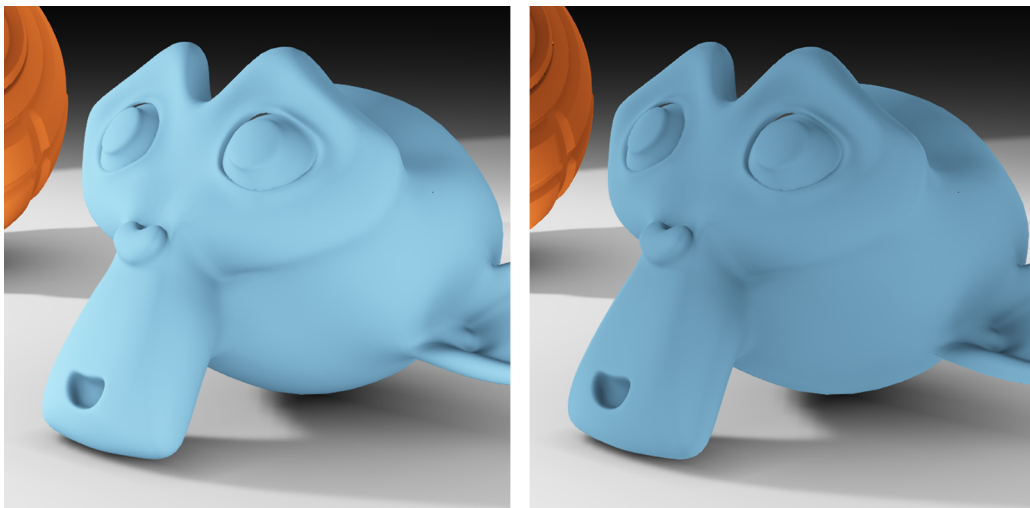


Abbildung 3.24 Eine diffuse Oberfläche nach Lambert (links) und eine rauere Oberfläche durch das Oren-Nayar-Modell (rechts)

3.5 Mehrschichtige Oberflächen

Die Kombination aus Diffus- und Glanzreflexion ermöglicht die Darstellung einzelner Materialien. Einige reale Oberflächen bestehen jedoch aus einer Kombination lichtdurchlässiger und lichtundurchlässiger Schichten mit unterschiedlichen Materialeigenschaften. Diese Schichten können unter anderem Lacke, Glasuren oder Kunststoffüberzüge sein.

Das Modell von Weidlich und Wilkie [WW07] stellt solche Oberflächen aus einer Kombination mehrerer BRDFs dar. Es modelliert sowohl die Absorption lichtdurchlässiger Beläge als auch die Möglichkeit der Totalreflexion in Abhängigkeit der Brechungsindizes der einzelnen Schichten (Abbildung 3.25).

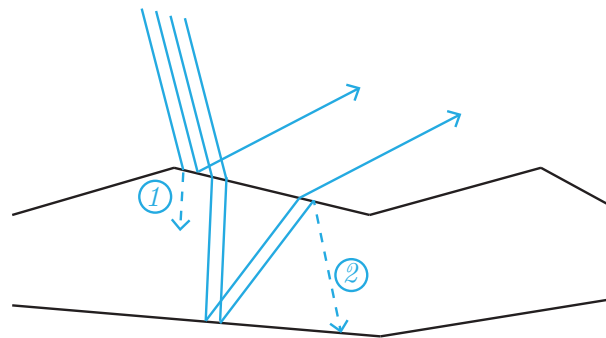


Abbildung 3.25 Teile des transmittierten Lichtes werden absorbiert (1) oder durch Totalreflexion in der Schicht eingefangen (2)

Um die sehr aufwändige Berechnung der Lichtstreuung in einem Medium zu vermeiden, werden folgende Vereinfachungen von Weidlich und Wilkie angenommen:

- Die Breite der Mikrofacetten ist weitaus größer als die Schichtdicke.
- Der Austrittspunkt des Lichts an einer Grenzfläche entspricht dem Eintrittspunkt.
- Die an einer Grenzfläche gebrochenen Strahlen treffen sich in einem Punkt.
- Licht wird ausschließlich an Grenzflächen durch Reflexion gestreut.

Das Modell ist also keine generelle Lösung, liefert aber innerhalb der Randbedingungen überzeugende Ergebnisse.

Die einzelnen Beläge einer Oberfläche können durch beliebige BRDFs dargestellt werden. Angesichts der geringen Auswirkung des Terms der internen Totalreflexion wird dieser, in der für diese Arbeit entwickelten Software, ebenso wie die Lichtbrechung an Grenzflächen aus Performancegründen vernachlässigt.

Somit ergibt sich das Modell aus [WW07], ohne die Möglichkeit der internen Totalreflexion, für eine Oberfläche mit zwei Schichten zu

$$f_W(\mathbf{p}, \omega_i, \omega_o) = f_{r1}(\mathbf{p}, \omega_i, \omega_o) + T_{12} f_{r2}(\mathbf{p}, \omega_i, \omega_o) e^{-\alpha d (\frac{1}{\omega_i} + \frac{1}{\omega_o})}. \quad (3.23)$$

T_{12} ist der Transmissionsgrad der, durch f_{r1} modellierten, oberen Schicht. Der Transmissionsgrad ergibt sich aus dem Reflexionsgrad F_{r1} dieser Schicht zu $T_{12} = 1 - F_{r1}$. Der Parameter α ist der wellenlängenabhängige Absorptionsgrad, d ist die Dicke des Belags.

Je flacher der Betrachtungswinkel, desto länger ist die in dem absorbierenden Belag zurückgelegte Strecke. So wird mehr Energie absorbiert, und weniger Licht, von der unteren Schicht ausgehend, erreicht den Betrachter.



Abbildung 3.26 *v.l.n.r.: Zweischichtige Materialien mit steigender Schichtdicke*

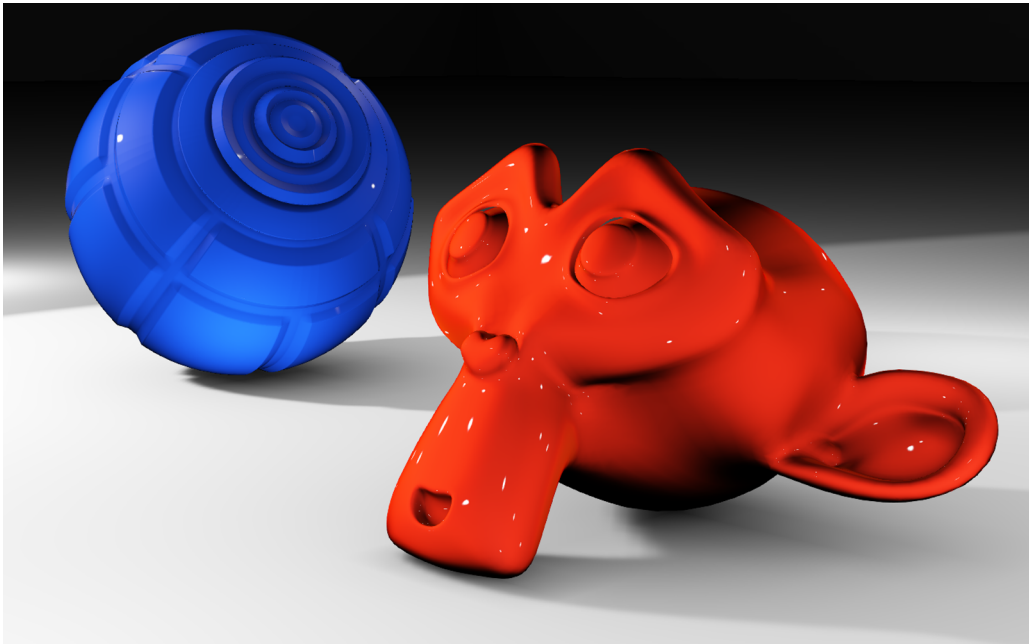


Abbildung 3.27 *Metall mit einer orangefarbenen Lackschicht*

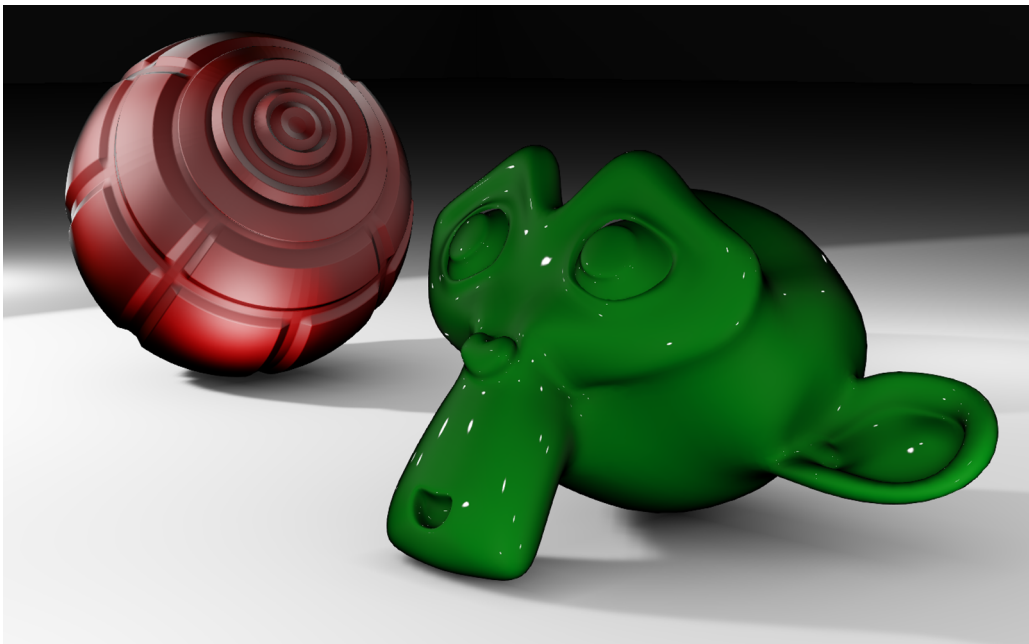


Abbildung 3.28 *diffus reflektierende Oberfläche mit einem absorbierenden Überzug*

4 Schatten

Bei der Bildsynthese durch Raytracing entstehen Schatten implizit durch lichtblockierende Geometrie. Bei der Beleuchtung einer rasterisierten Szene wird jedes Fragment unabhängig voneinander behandelt, globale Verdeckungsinformationen müssen anhand eines Schattierungsverfahrens gesondert erzeugt und bereitgestellt werden. Dabei wird durch die Schattierung nicht das gerenderte Bild abgedunkelt, sondern lediglich das Licht der verdeckten Lichtquelle abgeschwächt.

Die zwei gebräuchlichsten Verfahren zur Erzeugung dieser Verdeckungsinformationen sind die 1977 von Frank Crow vorgestellten *Shadow Volumes* [Cro77] sowie das 1978 von Lance Williams eingeführte *Shadow Mapping* [Wil78].

Bei dem Schattenwurf durch Shadow Volumes wird aus der Lichttrichtung und der Silhouette eines Objekts ein Volumen aufgestellt. Dieses Volumen wird als Geometrie gerendert. Pro Bildfragment kann nun bestimmt werden ob ein Punkt der Szene im Schatten liegt (Abbildung 4.1).

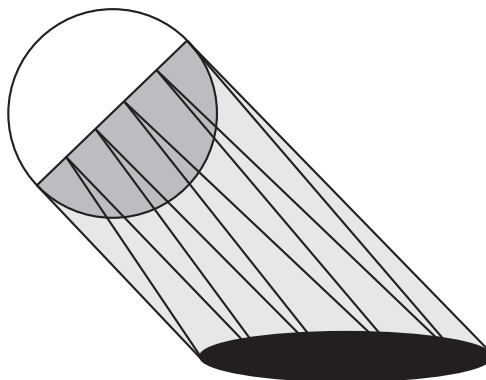


Abbildung 4.1 Ein Schattenvolumen durch Extrusion der Silhouette

Shadow Volumes erzeugen stets pixelgenaue Schatten. Die Performance ist jedoch abhängig von der Geometrie der Szene sowie der Position des Betrachters. Aufgrund des potenziell sehr hohen *Overdraws*, dem häufigen Überschreiben eines Fragments, wird anstelle von Shadow Volumes überwiegend das Shadow Mapping Verfahren eingesetzt.

4.1 Shadow Mapping

Shadow Mapping ist eine sehr effiziente Technik zur Erzeugung hardwarebeschleunigten Schattenwurfs in Echtzeitanwendungen.

Zuerst wird pro Bildfragment aus der Sicht der Lichtquelle der Tiefenwert z der Szene in eine Textur, die Shadow Map, gerendert (Abbildung 4.2).

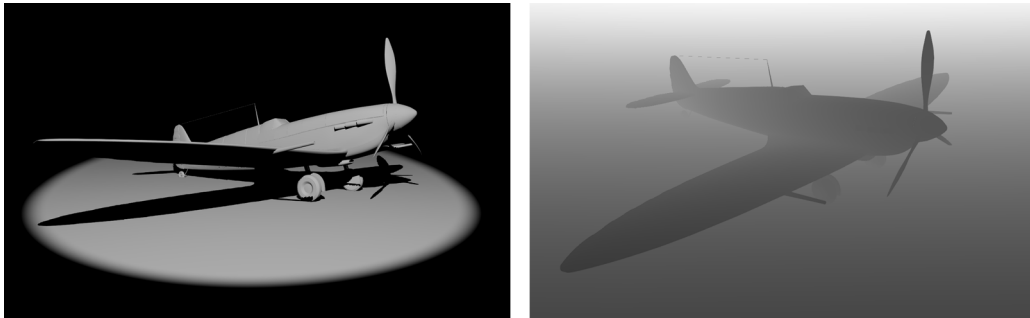


Abbildung 4.2 *Ein durch Shadow Mapping erzeugter Schatten (links) und die in der Shadow Map gespeicherten Werte (rechts)*

Ist ein zu beleuchtender Punkt p weiter von der Lichtquelle entfernt als der in der Shadow Map gespeicherte Wert z , dann wird das Licht durch Geometrie zwischen Lichtquelle und dem Punkt blockiert (Abbildung 4.3).

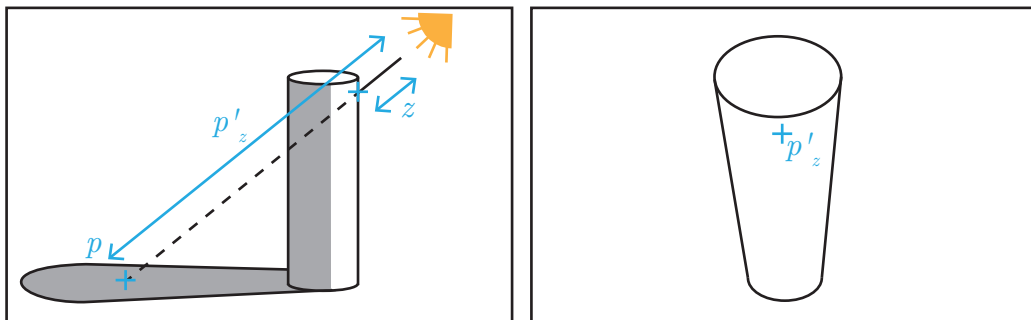


Abbildung 4.3 *Visualisierung eines verdeckten Punktes p (links) und die Projektion des Punktes auf die Bildebene der Lichtquelle (rechts)*

Die Werte einer Textur lassen sich, durch die API-spezifische Funktion $f_{texture}(\mathbf{c}_{tex})$ der verwendeten Grafik-Bibliothek, auslesen. Um den entsprechenden Tiefenwert z in der Shadow Map zu evaluieren, muss die Texturkoordinate \mathbf{c}_{tex} errechnet werden. Hierzu wird zunächst der Punkt auf die Bildebene der Lichtquelle abgebildet:

$$\mathbf{p}' = \mathbf{M}_{light} \mathbf{p} \quad (4.1)$$

Die Matrix \mathbf{M}_{light} beinhaltet sowohl die Transformation in das Koordinatensystem der Lichtquelle als auch die bei der Erzeugung der Shadow Map verwendete Projektion. Die perspektivische Division muss manuell durchgeführt werden, sofern dies nicht in Hardware übernommen wird:

$$\mathbf{p}' = \frac{1}{p'_w} \begin{pmatrix} p'_x \\ p'_y \\ p'_z \\ p'_w \end{pmatrix} \quad (4.2)$$

Letztendlich ergibt sich \mathbf{c}_{tex} durch Konvertierung der normalisierten Gerätekoordinaten in Texturkoordinaten zu

$$\mathbf{c}_{tex}(\mathbf{p}') = \begin{pmatrix} 0.5 + 0.5p'_x \\ 0.5 + 0.5p'_y \end{pmatrix}. \quad (4.3)$$

Der Schattenfaktor $\{s \in \mathbb{R} \mid 0 \leq s \leq 1\}$ zur Skalierung der Strahldichte der Lichtquelle lautet dann mit $z = f_{texture}(\mathbf{c}_{tex})$

$$s(p'_z, z) = \begin{cases} 0, & \text{falls } |p'_z| > z \\ 1, & \text{sonst} \end{cases}. \quad (4.4)$$

Die Strahlungsleistung Φ_i in Gleichung 2.7 aus Kapitel 2.2 ist, bei der Verwendung einer Shadow Map, die mit dem Schattenfaktor $s(p'_z, z)$ gewichtete Strahlungsleistung der Lichtquelle Φ_{light} :

$$\Phi_i = s(p'_z, z) \Phi_{light} \quad (4.5)$$

Das Shadow Mapping Verfahren weist aufgrund numerischer Präzision, sowie der Diskretisierung der Oberflächen einer Szene in einzelne Texel der Shadow Map, erhebliche Artefakte auf (Abbildung 4.4).

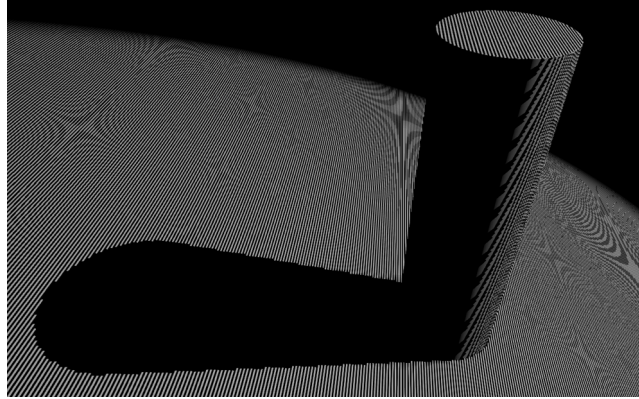


Abbildung 4.4 Artefakte durch Quantisierungsfehler einer Shadow Map

Jeder Texel der Shadow Map enthält einen einzigen Tiefenwert, deckt jedoch mehr als nur einen Punkt der Szene ab. Wird der Tiefenvergleich in Gleichung 4.4 für einen von der Lichtquelle sichtbaren Punkt p_A (Abbildung 4.5) durchgeführt, kann es durch Quantisierungsfehler zu Selbstverschattung auf der Oberfläche kommen.

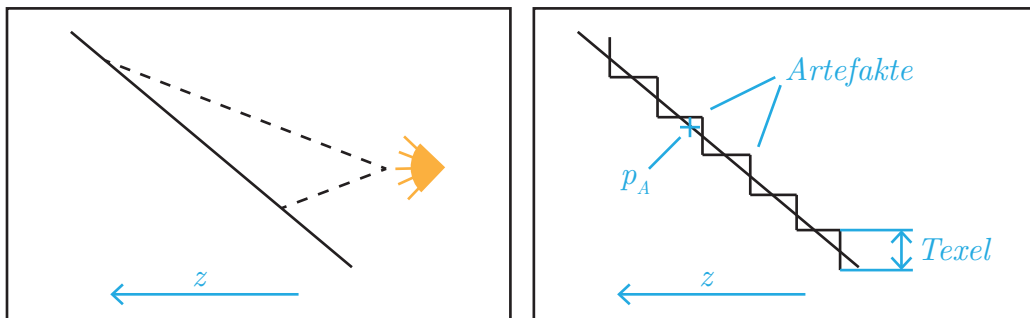


Abbildung 4.5 Ist eine Oberfläche nicht parallel zu der Bildebene der Shadow Map (links), kommt es zu Artefakten durch die Diskretisierung der Oberfläche (rechts)

Diesen Artefakten kann durch Addition eines konstanten, oder von der Ausrichtung der Oberfläche abhängigen, Tiefenwertes entgegengewirkt werden (Abbildung 4.6).

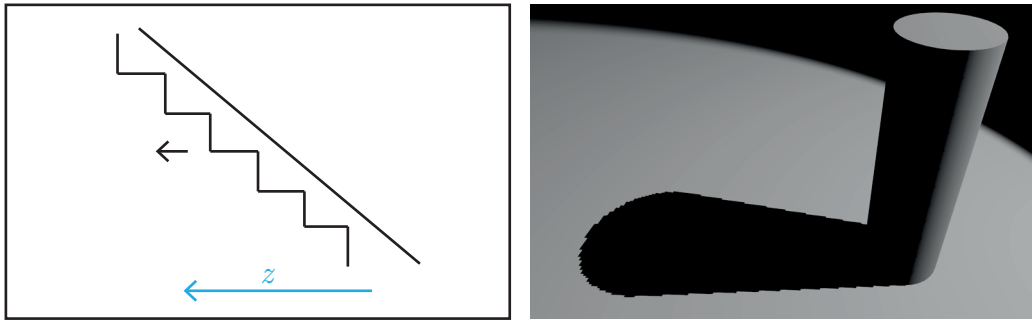


Abbildung 4.6 *Ein Möglichkeit zur Verringerung der Artefakte ist die Verschiebung der Tiefenwerte durch Addition einer Tendenz. Ist die Tendenz jedoch zu groß, entfernt sich der Schatten von der Geometrie*

Die oftmals einfachere Lösung ist es, die Tiefenwerte der Lichtquelle abgewandter Oberflächen in die Shadow Map zu rendern. Somit werden die der Lichtquelle zugewandten Primitive nie aufgrund von Quantisierungsfehlern schattiert. Auf den abgewandten Oberflächen werden die Fehler in der Regel durch das Lambertsche Kosinusetz (Kapitel 2.1.2) maskiert.

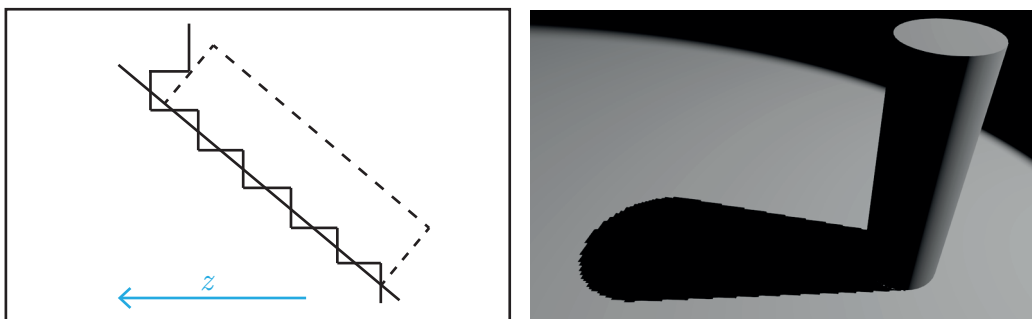


Abbildung 4.7 *Alternativ können Quantisierungsfehler durch das Rendern der Lichtquelle abgewandter Primitive verborgen werden. Hier kann es jedoch zur Beleuchtung eigentlich verdeckter Punkte an Kontaktstellen der Geometrie kommen*

4.2 Percentage Closer Filtering

Das von [Wil78] beschriebene Shadow Mapping Verfahren erzeugt durch den binären Schattierungsfaktor sehr harte Schattenränder.

Für ideale Punktlichtquellen ist dies unter Vernachlässigung der Lichtbeugung an Hindernissen zwar korrekt, jedoch sind ideale Punktlichtquellen technisch nicht realisierbar. Daher wirft jede reale Lichtquelle einen mehr oder weniger weichen Schatten.

Bei herkömmlichem Shadow Mapping werden zudem Aliasing-Effekte durch die endliche Auflösung der Shadow Map sichtbar (Abbildung 4.8).

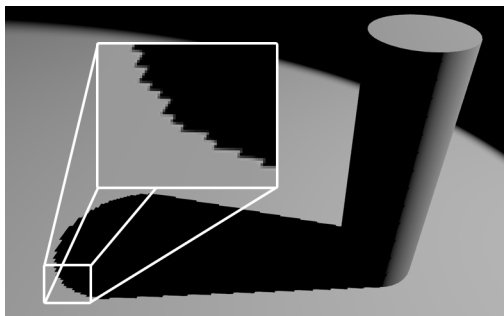


Abbildung 4.8 *Abhängig von Betrachtungsabstand und Auflösung der Shadow Map können Aliasing-Effekte ohne Percentage Closer Filtering deutlich sichtbar werden*

Reeves stellte 1987 das *Percentage Closer Filtering* (PCF) vor [RSC87], ein Verfahren zur Erzeugung weicher Schattenränder. Zur Filterung eines Schattenrandes werden mehrere Ergebnisse des binären Tiefenvergleichs (Gleichung 4.4) der in der Shadow Map um \mathbf{c}_{tex} liegenden Tiefenwerte z_k gemittelt (Abbildung 4.9):

$$s_{PCF}(p'_z, z) = \frac{1}{N} \sum_{k=1}^N s(p'_z, z_k) \quad (4.6)$$

Die Anzahl N sowie die Lage der umliegenden Tiefenwerte z_k ist frei wählbar.

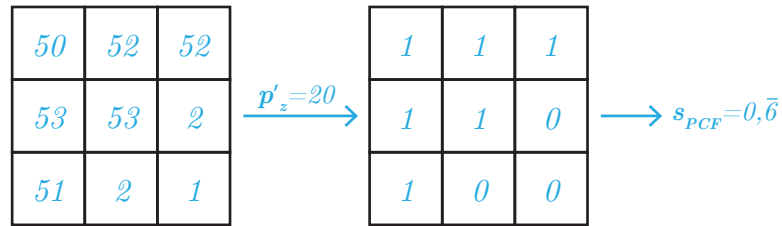


Abbildung 4.9 Die Tiefenwerte einer Shadow Map um die Texturkoordinate c_{tex} (links) werden mit p'_z verglichen (Mitte), der Mittelwert der Ergebnisse ist der Schattenfaktor s_{PCF} (rechts)

Je mehr Tiefenwerte verglichen werden, desto weicher ist der resultierende Schatten (Abbildung 4.10). Eine zu große Anzahl evaluierter Tiefenwerte kann sich jedoch aufgrund der begrenzten Speicherbandbreite einer Grafikkarte negativ auf die Performance auswirken.

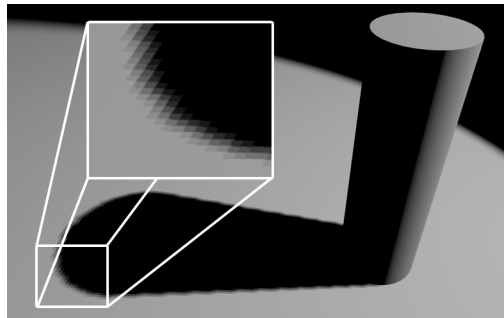


Abbildung 4.10 Ein durch PCF mit 9 Tiefenwerten gefilterter Schatten

Zur weiteren Steigerung der Qualität können die Ergebnisse des binären Schattenvergleichs bilinear interpoliert werden (Abbildung 4.11). Dieser Vorgang kann von einigen Grafikkarten hardwareseitig durchgeführt werden.

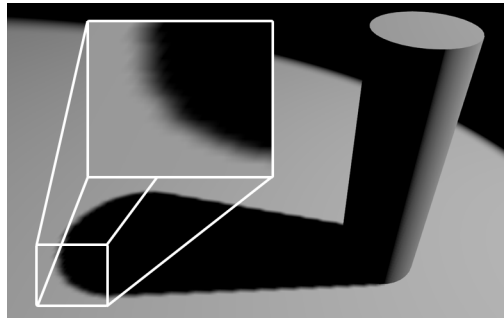


Abbildung 4.11 *Bilinear interpolierte, durch PCF gemittelte Schattenwerte bieten einen weichen Schattenrand mit wenig Aliasing-Effekten*

Aufgrund der hohen Empfindlichkeit des visuellen Systems in dunklen Bildbereichen kann eine bessere Verteilung der Graustufen eines gefilterten Schattens erreicht werden, wenn die Gamma-Vorentzerrung (Kapitel 2.4) nicht auf den Schattenfaktor angewendet wird (Abbildung 4.12).

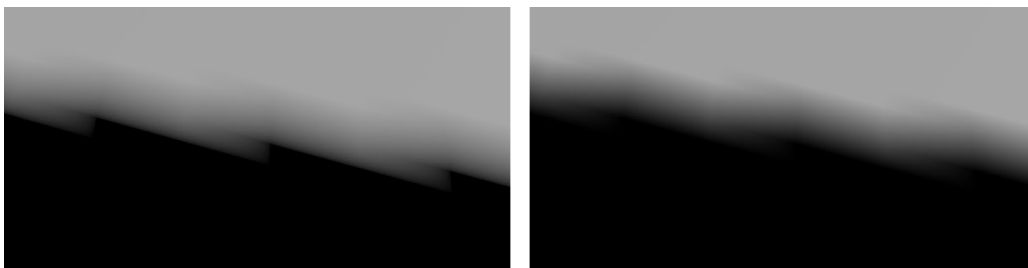


Abbildung 4.12 *Schatten mit Gamma-Vorentzerrung (links) und ohne Gamma-Vorentzerrung (rechts). Bei wenigen Vergleichswerten kann ohne Vorentzerrung ein besseres Ergebnis erzielt werden*

Durch den geringen Speicherbedarf sowie die Einfachheit der Implementierung ist Shadow Mapping, in Verbindung mit Percentage Closer Filtering, ein sehr verbreitetes Verfahren zur Erzeugung von Verdeckungsinformationen.

4.3 Variance Shadow Maps

Die von Donnelly und Lauritzen vorgestellten Variance Shadow Maps [DL06] ermöglichen die direkte Filterung der Shadow Map anstelle der Filterung der Ergebnisse des Tiefenvergleichs (Kapitel 4.2).

Die Tiefenwerte der Textur werden zunächst als ein Feld von Zufallswerten betrachtet. Das Ergebnis des Percentage Closer Filtering wird als Wahrscheinlichkeit der Sichtbarkeit durch die Lichtquelle innerhalb der Filterregion ausgedrückt:

$$s_{VSM}(p'_z, z) = P(z \geq |p'_z|) \quad (4.7)$$

Die einseitige Tschebyschow-Ungleichung bietet eine Obergrenze dieser Wahrscheinlichkeit mit $p_{max} \approx P(z \geq |p'_z|)$ für $\mu < |p'_z|$. Es gilt

$$p_{max} = \frac{\sigma^2}{\sigma^2 + (|p'_z| - \mu)^2} \quad (4.8)$$

mit dem Erwartungswert $\mu = E(z)$ sowie der Varianz $\sigma^2 = E(z^2) - E(z)^2$.

Der Erwartungswert $E(x)$ einer Zufallszahl x ist der gewichtete Mittelwert aller zufälligen Werte von x . Dieser gewichtete Mittelwert kann bei einer Variance Shadow Map durch beliebige Weichzeichnung der Textur erzeugt werden.

Wird also bei dem Rendervorgang der Shadow Map neben dem Tiefenwert z noch das Quadrat des Tiefenwertes z^2 gespeichert, kann p_{max} nach Gleichung 4.8 anhand einer gefilterten Textur bestimmt werden. Somit ergibt sich der Schattenfaktor einer Variance Shadow Map zu

$$s_{VSM}(p'_z, \mu, \sigma^2) = \begin{cases} 1, & \text{falls } |p'_z| \leq \mu \\ p_{max}, & \text{sonst} \end{cases} \quad (4.9)$$

Ein Verfahren zur Vermeidung von Quantisierungsfehlern bei der Generierung von Variance Shadow Maps wird in [Ngu07] beschrieben.

Die Filterbarkeit einer Variance Shadow Map ermöglicht neben hardwareseitigem Multisampling auch die Verwendung von Mipmaps sowie trilinearer und anisotroper Texturfilterung (Abbildung 4.13).

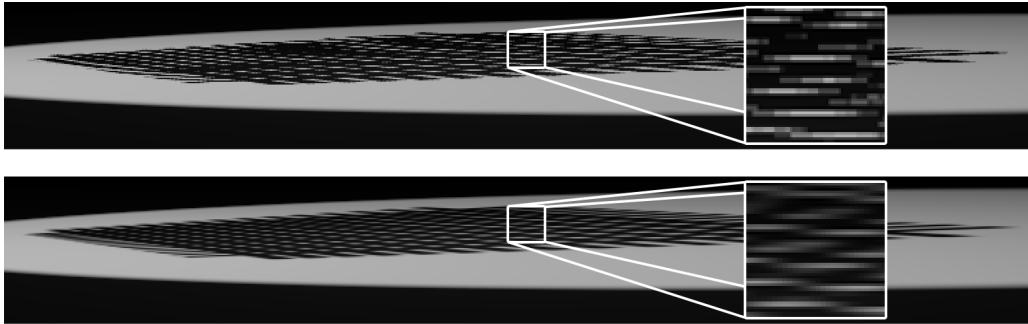


Abbildung 4.13 Vergleich einer Shadow Map mit bilinear interpoliertem PCF (oben) und anisotrop gefilterter Variance Shadow Map (unten). Durch die Texturfilterung werden Verkleinerungsartefakte vermieden

Ein weiterer Vorteil ist die Möglichkeit separierbare Filter anzuwenden. So lässt sich ein Filter über eine Region von $m \times n$ Texeln mit $m + n$ ausgelesenen Werten durch eine Hintereinanderausführung von zwei eindimensionalen Filtern sehr effizient umsetzen (Abbildung 4.14).

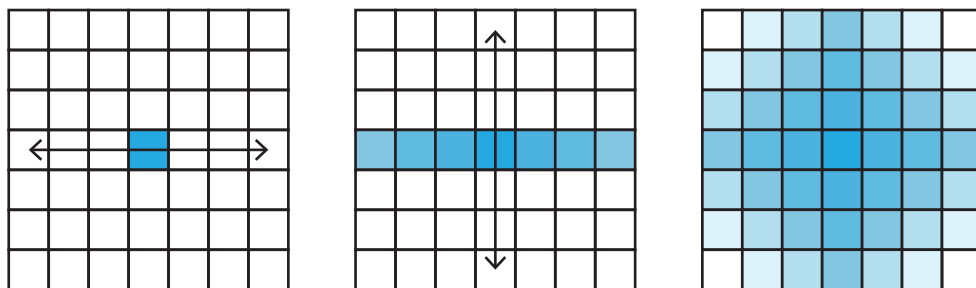


Abbildung 4.14 Bei einem separierbaren Filter wird eine zweidimensionale Filterung durch zwei eindimensionale Filter ersetzt

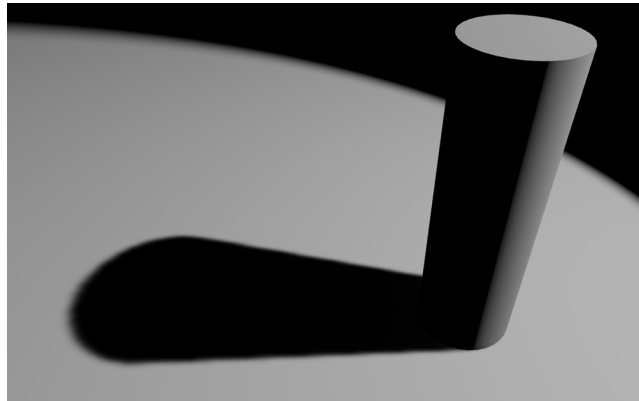


Abbildung 4.15 Eine mit einem 5×5 Gauß-Filter gefilterte Variance Shadow Map. Aliasing-Effekte sind nicht mehr erkennbar

Variance Shadow Maps bieten mit deutlich weniger Textursamples eine vergleichbare bis höhere Qualität (Abbildung 4.15) als normale Shadow Maps in Kombination mit Percentage Closer Filtering, sind jedoch nicht frei von Artefakten. Besonders bei einer hohen Tiefenkomplexität der Szene kann p_{max} aufgrund der großen Varianz innerhalb der Filterregion stark von dem exakten Wert der Wahrscheinlichkeit $P(z \geq |p'_z|)$ abweichen.

Die resultierenden Artefakte manifestieren sich in aufgehellten Umrissen lichtblockierender Geometrie innerhalb des Schattens (Abbildung 4.16).

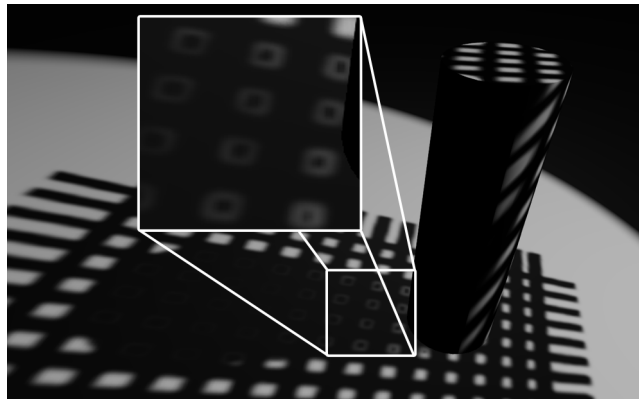


Abbildung 4.16 Artefakte bei Variance Shadow Maps durch eine hohe Tiefenkomplexität der Filterregion. Es erscheinen helle Umrisse in den Schatten

4.4 Exponential Variance Shadow Maps

Lauritzen zeigt in [LM08], dass die gravierendsten Artefakte bei Variance Shadow Maps auftreten, wenn das Verhältnis der Tiefenwerte mehrerer, verdeckender Geometrien innerhalb der Filterregion sehr groß ist. Um eine bessere Approximation der Wahrscheinlichkeit $P(z \geq |p'_z|)$ zu erreichen, kann dieses Verhältnis durch Verzerrung der Tiefenwerte verringert werden.

Für Exponential Variance Shadow Maps werden zwei exponentielle Verzerrungen angewendet, die jeweils unterschiedliche Artefakte verringern [LM08]:

$$\begin{aligned} z_p &= e^{cz} \\ z_n &= -e^{-cz} \end{aligned} \tag{4.10}$$

Die Konstante c skaliert den Grad der Verzerrung. Je größer c , desto stärker die Verzerrung und umso weniger Artefakte treten auf. Die Konstante sollte so groß wie möglich gewählt werden, jedoch klein genug sein um Präzisionsprobleme sowie einen arithmetischen Überlauf des verwendeten Datentyps zu vermeiden.

Bei dem Rendervorgang einer Exponential Variance Shadow Map werden dann die verzerrten Werte z_p , z_p^2 , z_n und z_n^2 in einer Vierkanal-Textur gespeichert. Um den Schattenfaktor zu bestimmen, muss der Tiefenwert p'_z des projizierten Punktes ebenfalls verzerrt werden.

Der Schattenfaktor einer Exponential Variance Shadow Map ergibt sich dann zu

$$s_{EVSM} = \min \{ s_{VSM}(e^{cp'_z}, \mu_p, \sigma_p^2), s_{VSM}(-e^{-cp'_z}, \mu_n, \sigma_n^2) \}. \tag{4.11}$$

Die Parameter μ_p , σ_p^2 , μ_n und σ_n^2 entsprechen den gefilterten, verzerrten Werten der Textur.

Nach [LM08] empfiehlt es sich die jeweilige Varianz auf eine Untergrenze σ_{min}^2 mit $\sigma^2 = \max \{ \sigma^2, \sigma_{min}^2 \}$ zu beschränken, um numerische Probleme für den Fall, dass der zu beleuchtende, verzerrte Tiefenwert dem Erwartungswert nahezu entspricht, zu vermeiden.

Die Verteilung der Tiefenwerte ist bei Exponential Variance Shadow Maps jedoch nicht linear. Daher ist der Minimalwert σ_{min}^2 nicht konstant, sondern abhängig von der nichtlinearen Änderung der Tiefenwerte.

Da die Varianz ein quadratisches Streuungsmaß ist, ist σ_{min}^2 also proportional zu dem Quadrat der Ableitung der Verzerrung. Die jeweilige Varianz in Gleichung 4.11 ist demnach

$$\sigma_p^2 = \max \{ \sigma_p^2, \sigma_{min}^2 (ce^{c|p_z|'})^2 \} \quad (4.12)$$

und

$$\sigma_n^2 = \max \{ \sigma_n^2, \sigma_{min}^2 (-ce^{-c|p_z|'})^2 \}. \quad (4.13)$$

Wird σ_{min}^2 zu groß gewählt, werfen kleine Tiefendifferenzen keinen Schatten. Wird die Untergrenze zu klein angesetzt, kommt es zu diffuser Selbstverschattung in den Ecken der Geometrie.

Exponential Variance Shadow Maps verbrauchen doppelt so viel Speicher wie Variance Shadow Maps, behalten jedoch alle Vorteile und weisen erheblich weniger Artefakte auf (Abbildung 4.17).

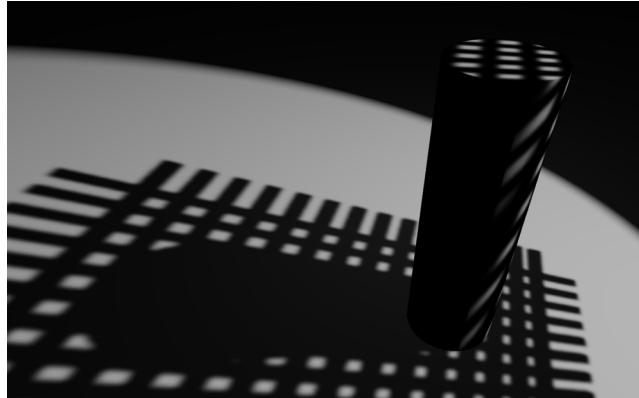


Abbildung 4.17 *Exponential Variance Shadow Maps weisen deutlich weniger Artefakte auf und bieten ein gutes Verhältnis von Qualität zu Rechenaufwand*

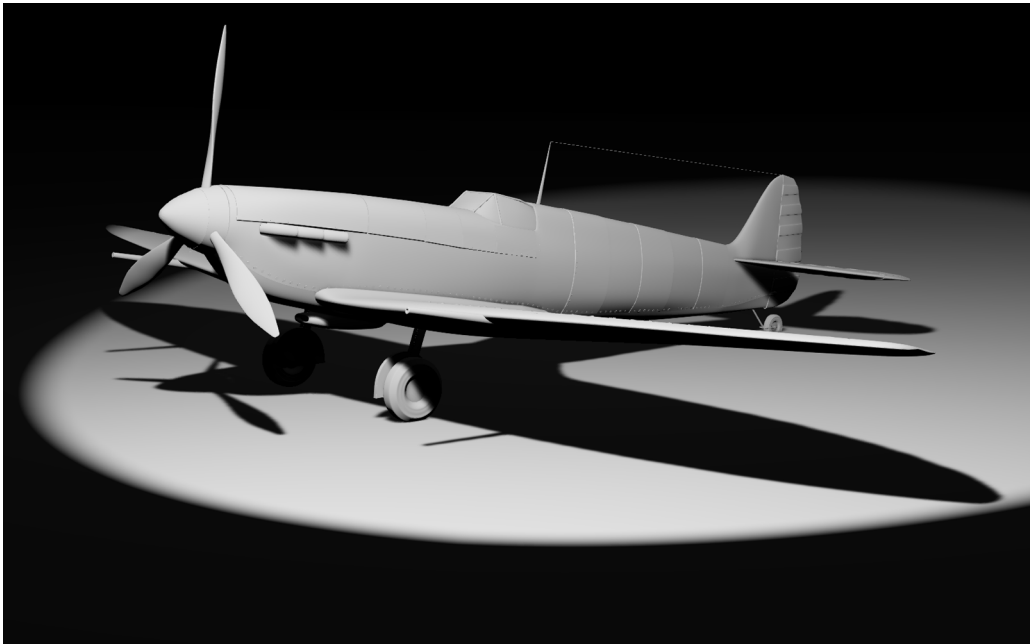


Abbildung 4.18 *Exponential Variance Shadow Map Beispiel 1*

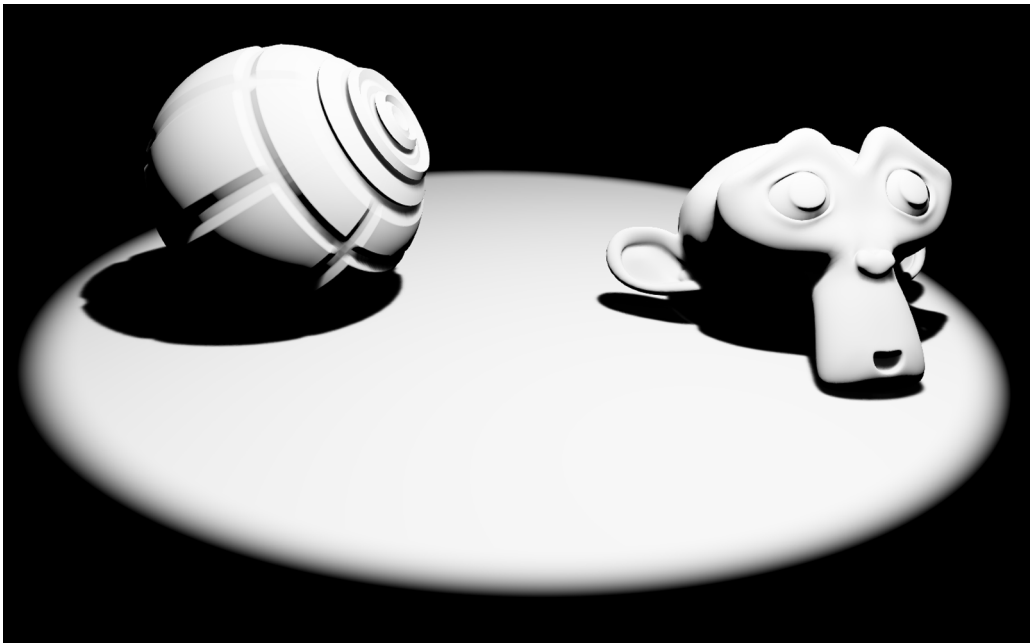


Abbildung 4.19 *Exponential Variance Shadow Map Beispiel 2*

4.5 Umgebungsverdeckung

In der Echtzeit-Computergrafik ist es aufgrund des Rechenaufwandes sowie Speicherbedarfs nicht praktikabel den Schatten jeder einzelnen Lichtquelle zu berechnen.

Um dennoch eine Schattierung der Oberflächen zu erreichen, kann eine Form der Umgebungsverdeckung (*engl. Ambient Occlusion*) verwendet werden [Lan02].

Die Umgebungsverdeckung $A(\mathbf{p})$ ist ein Maß für die Blockierung der Hemisphäre eines Punktes \mathbf{p} durch die umliegende Geometrie. Es gilt

$$A(\mathbf{p}) = \frac{1}{\pi} \int_{\Omega} V(\mathbf{p}, \omega) \cos \theta \, d\omega \quad (4.14)$$

mit $V(\mathbf{p}) = 0$ wenn die Richtung ω auf der Hemisphäre von \mathbf{p} durch Geometrie verdeckt ist, und $V(\mathbf{p}) = 1$ wenn in Richtung ω keine Verdeckung stattfindet. Der Winkel θ entspricht dem Winkel zwischen der Oberflächennormalen in \mathbf{p} und der Richtung ω .

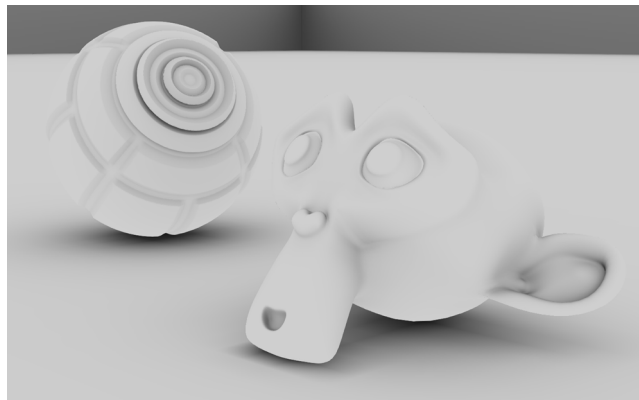


Abbildung 4.20 *Die Umgebungsverdeckung einer statischen Szene. Die Verdeckung wurde mit Blender vorberechnet und in Texturen gespeichert. Zur Laufzeit kann der Verdeckungsfaktor bei der Lichtberechnung evaluiert werden*

Die Lösung des Integrals in Gleichung 4.14 durch numerische Integration in hoher Qualität erweist sich auf moderner Hardware als nicht praktikabel. Eine Vielzahl

aktueller Forschungsarbeiten bieten verschiedene Approximationen dieser Gleichung an. Für statische Geometrie ist es jedoch aufgrund der deutlich höheren Qualität empfehlenswert, den Verdeckungsfaktor vorauszuberechnen (Abbildung 4.20).

Die Umgebungsverdeckung $A(\mathbf{p})$ kann bei der Beleuchtung einer Szene, wie auch der Schattenfaktor (Gleichung 4.5), zur Skalierung der Strahlungsleistung einer Lichtquelle verwendet werden (Abbildung 4.21). Es ist jedoch zu beachten, lediglich die Strahlungsleistung von Lichtquellen ohne Schattenwurf abzuschwächen.

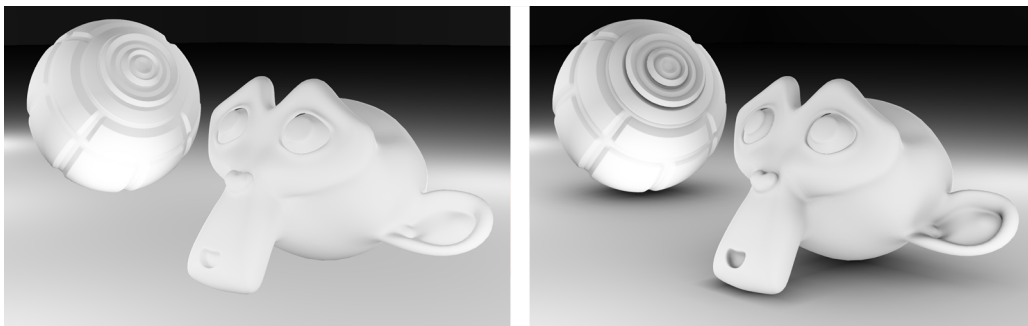


Abbildung 4.21 *Beleuchtung einer Szene mit mehreren Lichtquellen ohne Schattenwurf: ohne Umgebungsverdeckung (links) und mit Umgebungsverdeckung (rechts)*

Um einen möglichst guten räumlichen Eindruck einer Szene zu bewirken, ist es wesentlich wichtiger überhaupt eine Form von Schattierung zu verwenden, als absolut korrekte Schatten zu berechnen.

Die Anwendung der Umgebungsverdeckung als Ersatz des Schattenfaktors ist physikalisch nicht korrekt, bietet jedoch ein äußerst plausibles Ergebnis.

5 Implementierung

Die vorgestellten Techniken zur Beleuchtung und Schattierung wurden im Rahmen dieser Arbeit in einer privat entwickelten 3D-Grafik-Engine umgesetzt. Die verwendete Grafik-API ist OpenGL. Beleuchtung sowie Schattenberechnung werden in GLSL-Shaderprogrammen auf der Grafikkarte durchgeführt.

Die komplette Beleuchtung erfolgt mit linearen, radiometrischen Größen (Kapitel 2.1). Eine Gamma-Vorentzerrung (Kapitel 2.4) geschieht als letzter Schritt vor der Darstellung des Bildes, direkt nach der Dynamikkompression (Kapitel 2.3).

Als Beleuchtungsmodell wird das Torrance-Sparrow-Modell (Kapitel 3.3.2) für Glanzlichtreflexionen verwendet, diffuse Reflexionen werden durch den Lambert-schen Reflektor (Kapitel 3.4.1) dargestellt. Die Gewichtung von diffuser und gerichteter Reflexion erfolgt durch den Reflexionsgrad anhand der Brechzahl (Kapitel 3.2) des Materials. Der Reflexionsgrad wird mit der Näherung von Schlick ermittelt (Gleichung 3.6). Zusätzlich besteht die Möglichkeit, zweischichtige Materialien darzustellen (Kapitel 3.5).

Für den Schattenwurf werden bis zu 16 Exponential Variance Shadow Maps (Kapitel 4.4) mit einer Auflösung von jeweils 512×512 Bildpunkten sowie vierfachem Multisampling und 16-facher anisotroper Texturfilterung verwendet. Nach dem Rendervorgang einer Shadow Map wird ein separierbarer 5×5 Gauß-Filter angewendet. Das Umgebungslicht, bestehend aus eine Reihe von Punktlichtquellen, wird durch einen in dem Programm Blender vorausberechneten Verdeckungsfaktor schattiert (Kapitel 4.5).

Die zugrundeliegende 3D-Grafik-Engine übernimmt die Verwaltung der Geometrie einer Szene und verarbeitet die Eingaben des Benutzers. Jedem Geometrieelement kann durch die Benutzeroberfläche ein Material zugewiesen werden, eine Reihe voreingestellter Materialien stehen zur Auswahl bereit. Zur Laufzeit können die Materialparameter frei verändert werden. Es besteht zudem die Möglichkeit, Lichtquellen zu setzen oder zu entfernen sowie die Kamera frei im Raum zu bewegen.

Aufgrund der komplexen Beleuchtung wird ein *Depth-Prepass* durchgeführt: Vor dem eigentlichen Rendervorgang werden lediglich die Tiefenwerte der Szene ausgegeben und somit der Z-Buffer gefüllt.

Anschließend wird die Grafik-Pipeline anhand der verwendeten Grafik-API so konfiguriert, dass auch Bildfragmente, deren Tiefenwerte den gespeicherten Werten entsprechen, gerendert werden. So wird garantiert, dass der rechenaufwändige Materialshader bei der erneuten Rasterisierung der Szene nur ein einziges Mal pro Bildfragment ausgeführt wird. Abbildung 5.1 zeigt den Rendervorgang.

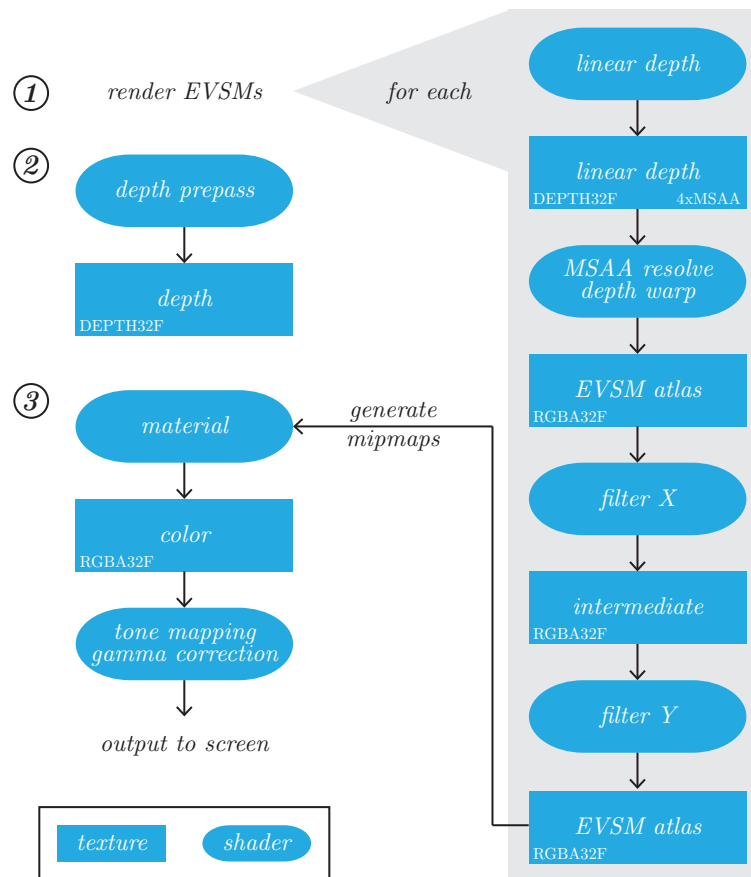


Abbildung 5.1 Zuerst werden alle Shadow Maps gerendert (1). Im Anschluss folgt der Depth Prepass (2) und letztendlich die Beleuchtung sowie Dynamikkompression mit Gamma-Vorentzerrung (3)

Um die Obergrenze der maximal gleichzeitig verwendbaren Texturen nicht zu überschreiten, werden alle Shadow Maps in einem Texturatlas gehalten [NVI04].

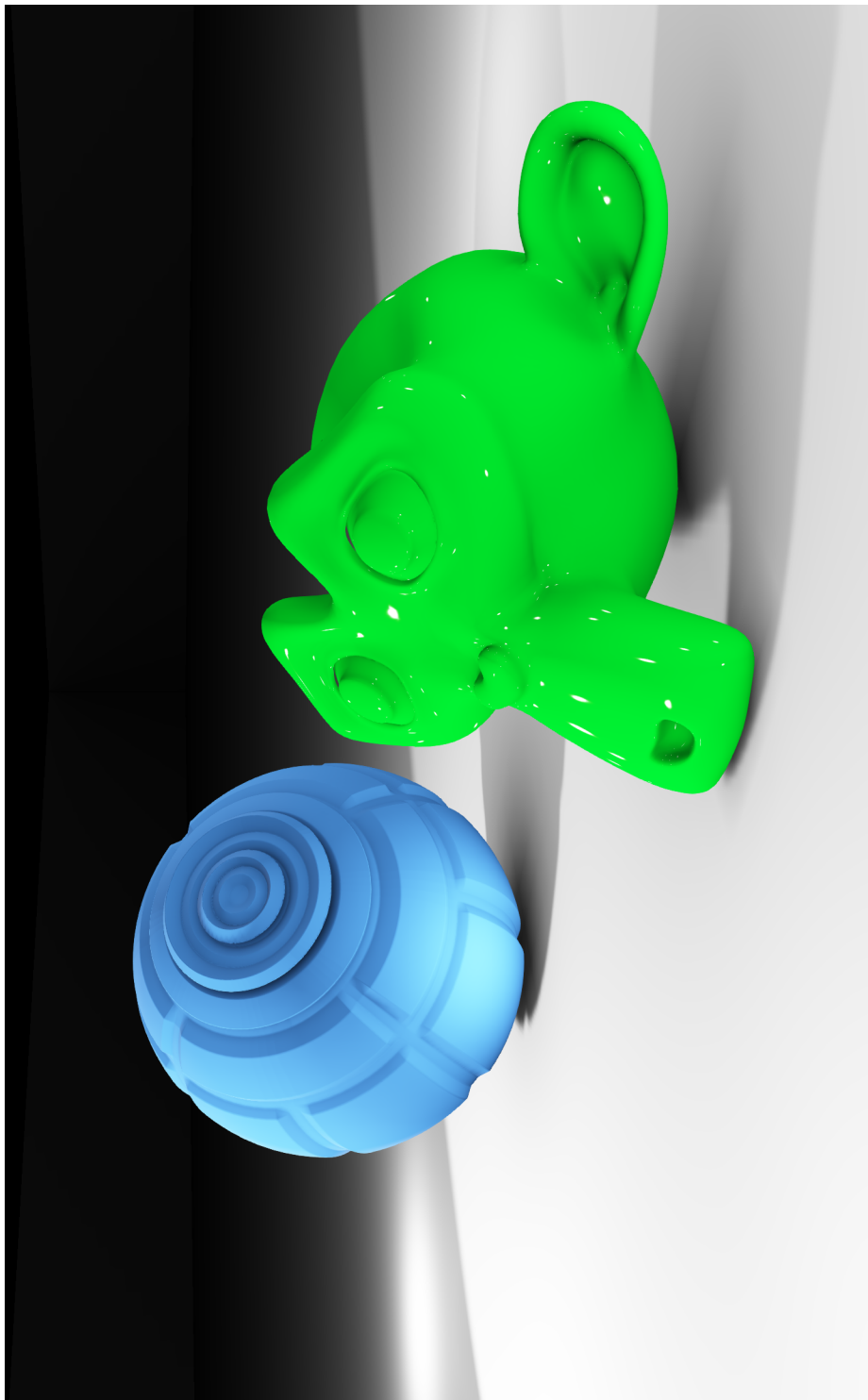
6 Zusammenfassung und Ausblick

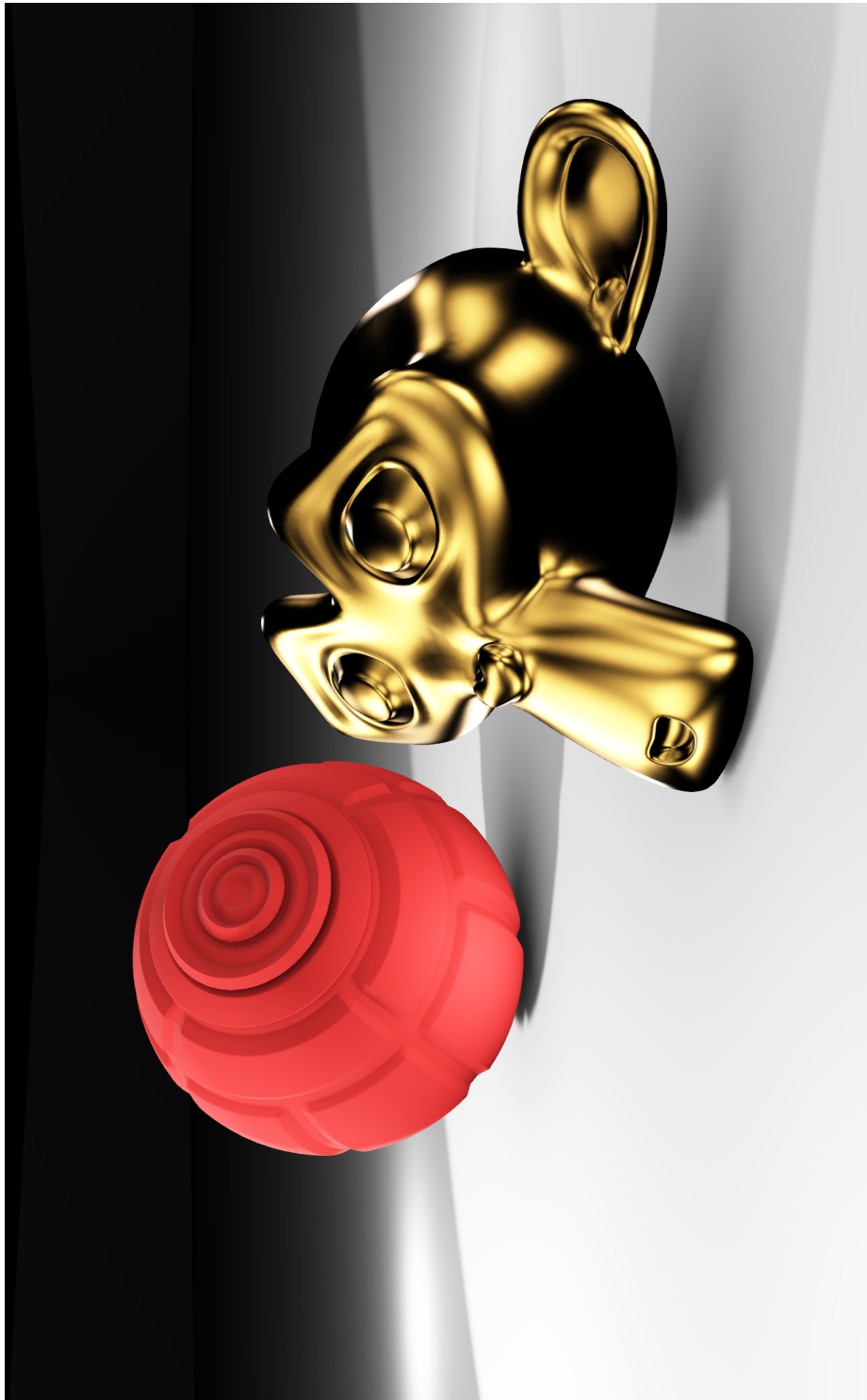
Das Ziel dieser Arbeit war die Umsetzung physikalisch plausibler Bildsynthese in einer Echtzeitumgebung.

Verschiedene Beleuchtungsmodelle der diffusen und gerichteten Lichtreflexion wurden verglichen und implementiert. Die physikalisch basierten Modelle konnten angesichts der hohen Vielseitigkeit der darstellbaren Materialien sehr überzeugende Ergebnisse liefern. Aufgrund der Vernachlässigung globaler Beleuchtungseffekte fällt jedoch bei stark reflektierenden Oberflächen die fehlende Reflexion der Umgebung deutlich auf. Die physikalisch plausible Umgebungsreflexion in Echtzeitanwendungen bietet eine interessante Aufgabenstellung für zukünftige Arbeiten.

Zusätzlich wurden verschiedene Verfahren zur Erstellung von Verdeckungsinformationen betrachtet. Zur Verringerung von Aliasing-Effekten des Schattenwurfs haben sich stochastische Methoden zur Ermessung des Grades der Schattierung, wenn auch nicht komplett frei von Artefakten, aufgrund des Verhältnisses von Qualität zu Rechenaufwand in Verbindung mit der linearen Filterbarkeit als vielversprechend erwiesen.

Insgesamt zeigt sich, dass durch die Verwendung realer, strahlungsphysikalischer Größen in Verbindung mit physikalisch plausiblen Beleuchtungsmodellen ein realitätsnaher Eindruck einer virtuellen Szene erreicht werden kann.





Literatur

- [AMHH08] AKENINE-MÖLLER, Tomas ; HAINES, Eric ; HOFFMAN, Natty: *Real-Time Rendering 3rd Edition*. Natick, MA, USA : A. K. Peters, Ltd., 2008. – 1045 S. – ISBN 987–1–56881–424–7
- [Bli77] BLINN, James F.: Models of light reflection for computer synthesized pictures. In: *SIGGRAPH Comput. Graph.* 11 (1977), Juli, Nr. 2, 192–198. <http://dx.doi.org/10.1145/965141.563893>. – DOI 10.1145/965141.563893. – ISSN 0097–8930
- [Cro77] CROW, Franklin C.: Shadow algorithms for computer graphics. In: *Proceedings of the 4th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM, 1977 (SIGGRAPH '77), 242–248
- [CT82] COOK, R. L. ; TORRANCE, K. E.: A Reflectance Model for Computer Graphics. In: *ACM Trans. Graph.* 1 (1982), Januar, Nr. 1, 7–24. <http://dx.doi.org/10.1145/357290.357293>. – DOI 10.1145/357290.357293. – ISSN 0730–0301
- [DL06] DONNELLY, William ; LAURITZEN, Andrew: Variance shadow maps. In: *Proceedings of the 2006 symposium on Interactive 3D graphics and games*. New York, NY, USA : ACM, 2006 (I3D '06). – ISBN 1–59593–295–X, 161–165
- [Gou71] GOURAUD, H.: Continuous Shading of Curved Surfaces. In: *IEEE Trans. Comput.* 20 (1971), Juni, Nr. 6, 623–629. <http://dx.doi.org/10.1109/T-C.1971.223313>. – DOI 10.1109/T-C.1971.223313. – ISSN 0018–9340
- [Ill04] ILLUMINATION, International C.: *Colorimetry*. Commission internationale de l'Eclairage, CIE Central Bureau, 2004 (CIE technical report). <http://books.google.de/books?id=P1NkAAAACAAJ>. – ISBN 9783901906336

- [Kaj86] KAJIYA, James T.: The rendering equation. In: *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM, 1986 (SIGGRAPH '86). – ISBN 0–89791–196–2, 143–150
- [KM00] KAUTZ, Jan ; MCCOOL, Michael D.: Approximation of Glossy Reflection with Prefiltered Environment Maps. In: *In Graphics Interface*, 2000, S. 119–126
- [LA92] LAMBERT, J.H. ; ANDING, E.: *Lamberts Photometrie: (Photometria, sive De mensura et gradibus luminis, colorum et umbrae) (1760)*. W. Engelmann, 1892 (Ostwalds Klassiker der exakten Wissenschaften). <http://books.google.de/books?id=Fq4RAAAAYAAJ>
- [Lan02] LANDIS, Hayden: Production-Ready Global Illumination. In: *Siggraph Course Notes* Bd. 16, 2002
- [LM08] LAURITZEN, Andrew ; MCCOOL, Michael: Layered variance shadow maps. In: *Proceedings of graphics interface 2008*. Toronto, Ont., Canada, Canada : Canadian Information Processing Society, 2008 (GI '08). – ISBN 978–1–56881–423–0, 139–146
- [LW94] LAFORTUNE, Eric P. ; WILLEMS, Yves D.: Using the Modified Phong Reflectance Model for Physically Based Rendering. 1994. – Forschungsbericht
- [NDM04] NGAN, Addy ; DURAND, Frédo ; MATUSIK, Wojciech: Experimental validation of analytical BRDF models. In: *ACM SIGGRAPH 2004 Sketches*. New York, NY, USA : ACM, 2004 (SIGGRAPH '04). – ISBN 1–58113–896–2, 90–
- [Ngu07] NGUYEN, Hubert: *Gpu gems 3*. First. Addison-Wesley Professional, 2007. – ISBN 9780321545428
- [Nic65] NICODEMUS, Fred E.: Directional Reflectance and Emissivity of an Opaque Surface. In: *Appl. Opt.* 4 (1965), Jul, Nr.

- 7, 767–773. <http://dx.doi.org/10.1364/AO.4.000767>. – DOI 10.1364/AO.4.000767
- [NVI04] NVIDIA: *SDK White Paper Improve Batching Using Texture Atlases*. 2004
- [ON94] OREN, Michael ; NAYAR, Shree K.: Generalization of Lambert’s reflectance model. In: *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM, 1994 (SIGGRAPH ’94). – ISBN 0–89791–667–0, 239–246
- [PH10] PHARR, Matt ; HUMPHREYS, Greg: *Physically Based Rendering, Second Edition: From Theory To Implementation*. 2nd. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 2010. – ISBN 0123750792, 9780123750792
- [Pho75] PHONG, Bui T.: Illumination for computer generated pictures. In: *Commun. ACM* 18 (1975), Juni, Nr. 6, 311–317. <http://dx.doi.org/10.1145/360825.360839>. – DOI 10.1145/360825.360839. – ISSN 0001–0782
- [PPB05] PEDROTTI, F. ; PEDROTTI, L. ; BAUSCH, W.: *Optik Für Ingenieure: Grundlagen*. Springer, 2005 <http://books.google.de/books?id=GTDiER12nbwC>. – ISBN 9783540228134
- [RSC87] REEVES, William T. ; SALESIN, David H. ; COOK, Robert L.: Rendering antialiased shadows with depth maps. In: *SIGGRAPH Comput. Graph.* 21 (1987), August, Nr. 4, 283–291. <http://dx.doi.org/10.1145/37402.37435>. – DOI 10.1145/37402.37435. – ISSN 0097–8930
- [Sch94] SCHLICK, Christophe: An Inexpensive BRDF Model for Physically-Based Rendering. In: *Comput. Graph. Forum* (1994), S. 233–246
- [Sny96] SNYDER, John M.: *Area Light Sources for Real-Time Graphics*. 1996

-
- [TS67] TORRANCE, K.E. ; SPARROW, E.M.: Theory for Off-Specular Reflection from Roughened Surfaces. 57 (1967), September, Nr. 9, S. 1105–1114
- [Wil78] WILLIAMS, Lance: Casting curved shadows on curved surfaces. In: *Proceedings of the 5th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM, 1978 (SIGGRAPH '78), 270–274
- [WW07] WEIDLICH, Andrea ; WILKIE, Alexander: Arbitrarily Layered Micro-Facet Surfaces. In: *GRAPHITE 2007*, ACM, Dezember 2007. – ISBN 978-1-59593-912-8, 171–178

Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, dass alle Stellen der Arbeit, die wörtlich oder sinngemäß aus anderen Quellen übernommen wurden, als solche kenntlich gemacht sind und dass die Arbeit in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegt wurde.

Köln, den 3. August 2012